TELEDYNE DALSA

Teledyne DALSA • 7075 Place Robert-Joncas, Suite 142 • St-Laurent, Quebec, H4M 2Z2 • Canada
www.teledynedalsa.com/mv

X64-CL Series
User's Manual

Part number OC-64CM-CLU00

Edition 2.00

NOTICE

© 2001-2011 TELEDYNE DALSA All rights reserved.

This document may not be reproduced nor transmitted in any form or by any means, either electronic or mechanical, without the express written permission of Teledyne DALSA. Every effort is made to ensure the information in this manual is accurate and reliable. Use of the products described herein is understood to be at the user's risk. Teledyne DALSA assumes no liability whatsoever for the use of the products detailed in this document and reserves the right to make changes in specifications at any time and without notice.

Microsoft® is a registered trademark; Windows®, Windows XP®, Windows Vista®, and Windows® 7 are trademarks of Microsoft Corporation.

All other trademarks or intellectual property mentioned herein belongs to their respective owners.

Edition 2.00 released on: June 21, 2011

Document Number: OC-64CM-CLU00

Printed in Canada

Contents

X64-CL SERIES OVERVIEW **5**
PRODUCT PART NUMBERS 5
ABOUT THE X64-CL SERIES OF FRAME GRABBERS 7
X64-CL Series Key Features 7
User Programmable Configurations 7
ACUPlus: X64-CL Acquisition Control Unit 8
DTE: Intelligent Data Transfer Engine 9
Advanced Controls Overview 9
DEVELOPMENT SOFTWARE OVERVIEW 10
Sapera LT Library 10
Sapera Processing Library 10

INSTALLING THE X64-CL **11**
WARNING! (GROUNDING INSTRUCTIONS) 11
INSTALLATION 11
Sapera LT Library Installation 12
X64-CL Driver Installation 12
X64-CL Firmware Loader 13
REQUIREMENTS FOR A SILENT INSTALL 15
Silent Mode Installation 15
Silent Mode Uninstall 16
Silent Mode Installation Return Code 17
Installation Setup with CorAppLauncher.exe 17
Custom Driver Installation using install.ini 18
UPGRADING SAPERA OR BOARD DRIVER 19
Board Driver Upgrade Only 19
Upgrading both Sapera and Board Driver 20
USING THE CAMERA LINK SERIAL CONTROL PORT 20
COM Port Assignment 21
Setup Example with Windows HyperTerminal 22
DISPLAYING X64-CL BOARD INFORMATION 23
Device Manager – Board Viewer 23
CONFIGURING SAPERA 24
Viewing Installed Sapera Servers 24
Increasing Contiguous Memory for Sapera Resources 24
Contiguous Memory for Sapera Messaging 25
TROUBLESHOOTING INSTALLATION PROBLEMS 26
Recovering from a Firmware Update Error 26
Windows Event Viewer 27

Teledyne DALSA Device Manager Program *27*
PCI Configuration *29*
Sapera and Hardware Windows Drivers *30*
Log Viewer *31*
Windows Device Manager *32*
Memory Requirements with Area Scan Acquisitions *33*
Symptoms: CamExpert Detects no Boards *33*
Symptoms: X64-CL Does Not Grab *34*
Symptoms: Card grabs black *34*
Symptoms: Card acquisition bandwidth is less than expected *34*

CAMEXPERT QUICK START FOR THE X64-CL 35
INTERFACING CAMERAS WITH CAMEXPERT 35
CamExpert Example with a Monochrome Camera *35*
CAMEXPERT DEMONSTRATION AND TEST TOOLS 37
CAMERA TYPES & FILES APPLICABLE TO THE X64-CL 37
Overview of Sapera Acquisition Parameter Files (.ccf or *.cca/*.cvi)* *38*
Saving a Camera File *39*
Camera Interfacing Check List *39*
LINESCAN EXAMPLE: INTERFACING THE TELEDYNE DALSA PIRANHA2 LINESCAN CAMERA 40
CamExpert Interfacing Outline *40*
Step 1: Piranha2 in Free Run Exposure Mode *41*
File Selection & Grab Test *41*
Overview of Basic Timing Parameters *42*
Step 2: Piranha2 in External Exposure Mode *44*
Advanced Control Parameters *44*
Step 3: Piranha2 with Shaft Encoder Line Sync *46*
Shaft Encoder Line Sync Setup *46*
Shaft Encoder with Fixed Frame Buffer Setup *47*
Shaft Encoder with Variable Frame Buffer Setup *48*

SAPERA DEMO APPLICATIONS 49
GRAB DEMO OVERVIEW 49
Using the Grab Demo *49*
Grab Demo Main Window *50*

X64-CL REFERENCE 51
X64-CL FULL BLOCK DIAGRAM 51
X64-CL DUAL BLOCK DIAGRAM 52
X64-CL ACQUISITION TIMING 53
EXTERNAL FRAME TRIGGER SOURCE SELECTION FOR AREA SCAN APPLICATION 54
Parameter Values Specific to the X64-CL *54*
LINE TRIGGER SOURCE SELECTION FOR LINESCAN APPLICATIONS 56
Parameter Values Specific to the X64-CL *56*
SHAFT ENCODER INTERFACE TIMING 58

VIRTUAL FRAME TRIGGER FOR LINESCAN CAMERAS 59
SAPERA ACQUISITION METHODS 61
TRIGGER-TO-IMAGE RELIABILITY 61
Supported Events and Transfer Methods 62
OUTPUT LUT AVAILABILITY 64
Firmware: Standard (default) 64
Firmware: X64-CL-Full - 12-bit LUT 65
SUPPORTING NON-STANDARD CAMERA LINK CAMERAS 65
Firmware: X64-CL 10-Tap Format 65
X64-CL SAPERA CAPABILITIES 67
Camera Related Capabilities 67
Camera Related Parameters 67
VIC Related Parameters 72
ACQ Related Parameters 76

X64-CL SAPERA SERVERS & RESOURCES 77
SERVERS AND RESOURCES 77

TECHNICAL SPECIFICATIONS 79
BOARD SPECIFICATIONS 79
HOST SYSTEM REQUIREMENTS 80
EMI CERTIFICATIONS 81
CONNECTOR AND SWITCH LOCATIONS 82
X64-CL Layout Drawings 82
Connector List (X64-CL half length board) 83
X64-CL EM Revision A2 Layout Drawing 84
Connector List (X64-CL EM full length board) 85
CONNECTOR AND SWITCH SPECIFICATIONS 85
X64-CL Connector View 85
Status LEDs Functional Description 86
J1: Camera Link Connector 1 (applies to X64-CL Dual & Full models) 87
J2: Camera Link Connector 2 (on X64-CL Dual model) 87
J2: Camera Link Connector 2 (on X64-CL Full model used with Medium or Full cameras) 88
Camera Link Camera Control Signal Overview 89
J4 (CL or EM rev. A0, A1): External Signals Connector 89
X64-CL or X64-EM rev. A0, A1: External Signals Connector Bracket Assembly 93
J201 (X64-EM rev. A2 & later): External Signals Connector 95
X64-EM rev. A2 & later: External Signals Connector Bracket Assembly 97
Hirose Connector Module Option (connects internally to J3) 99
LVDS Shaft Encoder Input Specifications 100
Connecting a TTL Shaft Encoder Signal to the LVDS Input 101
External Trigger TTL Input Electrical Specification 102
Strobe TTL Output Electrical Specification 104
J8 (CL), J12 (CL EM): Power to Camera Voltage Selector 105

J9 (CL), J13 (CL EM): PC Power to Camera Interface *106*
J11 (CL), J14 (CL EM): Start Mode *106*
J7, J12 (CL): Reserved *106*
J7, J9, J10, J11, J15, J200, J202 (CL EM): Reserved *106*
Brief Description of Standards RS-232, RS-422, & RS-644 (LVDS) *107*

CAMERA LINK INTERFACE **109**
CAMERA LINK OVERVIEW 109
Rights and Trademarks *109*
DATA PORT SUMMARY 110
CAMERA SIGNAL SUMMARY 110
CAMERA LINK CABLES 111

CONTACT INFORMATION **113**
SALES INFORMATION 113
TECHNICAL SUPPORT 114

GLOSSARY OF TERMS **115**

INDEX **119**

X64-CL Series Overview

Product Part Numbers

X64-CL Board Family

Product	Product Number	Product Number (RoHS compliant)
66MHz Pixel Clock X64-CL Full with 32 MB of memory X64-CL Full with 128 MB of memory	 OC-64C0-00080[1] OC-64C0-02080[1]	 OR-64C0-00080[1] OR-64C0-02080[1]
85MHz Pixel Clock X64-CL Full with 32 MB of memory X64-CL Full with 128 MB of memory X64-CL Full with 256 MB of memory X64-CL Full with 1 GB of memory X64-CL Full with 2 GB of memory	 OC-64C0-10080[1] OC-64C0-12080[1] OC-64C0-13080[2] OC-64C0-15080[2] OC-64C0-16080[2]	 OR-64C0-10080[1] OR-64C0-12080[1]
66MHz Pixel Clock X64-CL Dual with 32 MB of memory X64-CL Dual with 128 MB of memory X64-CL Dual with 256 MB of memory	 OC-64C0-00060[1] OC-64C0-02060[1]	 OR-64C0-00060[1] OR-64C0-02060[1]
85MHz Pixel Clock X64-CL Dual with 32 MB of memory X64-CL Dual with 128 MB of memory X64-CL Dual with 256 MB of memory X64-CL Dual with 1 GB of memory X64-CL Dual with 2 GB of memory	 OC-64C0-10060[1] OC-64C0-12060[1] OC-64C0-13060[2] OC-64C0-15060[2] OC-64C0-16060[2]	 OR-64C0-10060[1] OR-64C0-12060[1]
Optional Hirose Connector Module (CMI #194) see "X64-CL Connector View" on page 85	Contact Sales at Teledyne DALSA Montreal	
Note 1: see "X64-CL Layout Drawings" on page 82 where various board revisions are shown. Note 2: see "X64-CL EM Revision A2 Layout Drawing" on page 84.		
For OEM clients this manual, in printed form, is available on request	OC-64CM-CLU00	

X64-CL Software

Item	Product Number
Sapera LT version 5.20 or later (required but sold separately) 1. Sapera LT: Provides everything needed to build imaging applications 2. Current Sapera compliant board hardware drivers 3. Board documentation in Adobe Acrobat® (PDF) 4. Sapera documentation in compiled HTML help, and PDF formats	OC-SL00-0000000
(optional) Sapera Processing Imaging Development Library includes over 600 optimized image processing routines.	Contact Sales at Teledyne DALSA Montreal

X64-CL Cables & Accessories

Item	Product Number
External Signals Connector Bracket Assembly supplied with each X64-CL and some X64-EM revisions (connects to J4 – see "X64-CL or X64-EM rev. A0, A1: External Signals Connector Bracket Assembly" on page 93)	OR-64CC-0TIO2
External Signals Connector Bracket Assembly supplied with each X64-EM revision A2 or later (connects to J201 – see "X64-EM rev. A2 & later: External Signals Connector Bracket Assembly" on page 97)	OR-64CC-GIO25EM
(optional) Power interface cable required when supplying power to cameras	OR-COMC-POW03
(optional) Camera Link Video Input Cable: 1 meter 2 meter	 OC-COMC-CLNK0 OC-COMC-CLNK6
(optional) DB25 male to color coded blunt end cable – 6 foot (1.82 meter) length	OR-COMC-XEND1

About the X64-CL Series of Frame Grabbers

X64-CL Series Key Features

- Two Models available:
- X64-CL Full: 1 Full or 1 Medium or 1 Base Camera
- X64-CL Dual: 1 or 2 Base Cameras, supporting:
  - Simultaneous capture from 2 Base cameras
  - Cameras do not need to be synchronized
  - Can mix monochrome and color cameras
- Targeted for general purpose machine vision applications
- Two Camera Link interface (MDR26) connectors
- Single slot
- 32MB-2GB frame buffer memory
- Trigger input
  - TTL or LVDS signal type is software selectable
  - Signal type must be the same for both camera inputs on the X64-CL Dual
- Quadrature Shaft encoder input (LVDS signal type only)
- Camera control signals
- Dual Strobe Outputs

User Programmable Configurations

Use the X64-CL firmware loader function from within the Teledyne DALSA Device manager utility to select firmware for one of the supported modes. Firmware selection is made either during driver installation or manually later on.

Important: Not all firmware versions are supported by each version or board revision of X64-CL or X64-CL EM. Contact Teledyne DALSA Montreal sales for the X64 version required.

Firmware versions currently available with the Full board version:

- **X64-CL Standard (default)**
  Support for up to 8-Tap cameras with 8-bit or 10-bit LUT (see "Firmware: Standard (default)" on page 64 for details on these LUT modes). See "X64-CL Sapera Capabilities" on page 67 for a complete listing of capabilities.
- **X64-CL 12-Bit LUT**
  Support for 12-Bit LUT with 12-Bit monochrome or 3x12-Bit RGB sources. This firmware offers the same functionality as the “Standard” one, except for the LUT support. See "Firmware: X64-CL-Full - 12-bit LUT" on page 65 for details on these LUT modes.
- **X64-CL 10-Tap Format** (not a Camera Link standard),
  Firmware for the X64-CL 85 MHz to support cameras such as the Basler A504K.
  See "Firmware: X64-CL 10-Tap Format" on page 65 for addition details.

Firmware version currently available with the Dual board:

- **Two independent Base Camera Link Input** (*installation default selection*)
  Support for one or two Base Camera Link cameras.

ACUPlus: X64-CL Acquisition Control Unit

- ACUPlus consists of two sets of independent grab controllers, one pixel packer, and one time base generator. ACUPlus delivers a flexible acquisition front end plus it supports pixel clock rates of up to 66MHz (85MHz optional).
- ACUPlus acquires variable frame sizes up to 256KB per horizontal line and up to 16 million lines per frame. ACUPlus can also capture an infinite number of lines from a linescan camera without losing a single line of data.
- ACUPlus supports standard Camera Link multi-tap configurations from 8 to 24-bit/pixels. Additionally, alternate tap configurations can support up to 8 taps of 8-bits each.
- X64-CL Dual supports two cameras with different tap configurations simultaneously such as one dual tap 8/10/12-bits monochrome camera with an RGB 24-bit camera.

Camera Link Maximum Acquisition Rates:

This table specifies the X64-CL acquisition hardware maximums, not the maximum data transfer rate through the PCI 64 bus to system memory.

Cameras connected	Camera Link standard	Maximum Acquisition rate 66 MHz components	Maximum Acquisition rate 85 MHz components
2	Base	396 Mbytes/sec	510 Mbytes/sec
1	Medium	396 Mbytes/sec	510 Mbytes/sec
1	Full	528 Mbytes/sec	680 Mbytes/sec

DTE: Intelligent Data Transfer Engine

The X64-CL intelligent Data Transfer Engine ensures fast image data transfers between the board and the host computer with zero CPU usage. The DTE provides a high degree of data integrity during continuous image acquisition in a non-real time operating system like Windows. DTE consists of multiple independent DMA units, Tap Descriptor Tables, and Auto-loading Scatter-Gather tables.

Advanced Controls Overview

Visual Indicators

X64-CL features two unique visual LED indicators to facilitate system installation and setup. These provide visual feedback indicating when the camera is connected properly and sending data.

External Event Synchronization

Two sets of dedicated trigger inputs (100ns minimum pulse width) and strobe signals are provided to synchronize precisely image captures with external events.

Camera Link Communications ports

Two PC independent communication ports provide Camera Link controls for camera configurations. These ports do not require addition PC resources like free interrupts or I/O address space. Accessible via the board device driver, the communication ports preset a seamless interface to Windows-based standard communication applications like HyperTerminal, etc. The communication ports are accessible directly from the Camera Link connectors.

Quadrature Shaft Encoder

Important feature for web scanning applications, the Quadrature-Shaft-Encoder inputs allow synchronized line captures from external web encoders.

PCI 64 bit Interface

The X64-CL is a universal PCI slot board, compliant with the PCI version 2.2 specification for 64 bit devices.

The X64-CL operates in:

- 64 bit – 66/33 MHz PCI slots
- 32 bit – 33 MHz PCI slots

Transfer rates up to 528 Mbytes/sec with the appropriate camera are possible. The X64-CL board occupies one computer expansion slot and one chassis opening.

Development Software Overview

Sapera LT Library

Sapera LT is a powerful development library for image acquisition and control. Sapera LT provides a single API across all current and future Teledyne DALSA hardware. Sapera LT delivers a comprehensive feature set including program portability, versatile camera controls, flexible display functionality and management, plus easy to use application development wizards.

Sapera LT comes bundled with CamExpert, an easy to use camera configuration utility to create new, or modify existing camera configuration files.

Sapera Processing Library

Sapera Processing is a comprehensive set of C++ classes for image processing and analysis. Sapera Processing offers highly optimized tools for image processing, blob analysis, search (pattern recognition), OCR and barcode decoding.

Installing the X64-CL

Warning! (Grounding Instructions)

Static electricity can damage electronic components. Please discharge any static electrical charge by touching a grounded surface, such as the metal computer chassis, before performing any hardware installation. If you do not feel comfortable performing the installation, please consult a qualified computer technician.

Important: Never remove or install any hardware component with the computer power on. Disconnect the power cord from the computer to disable the power standby mode. This prevents the case where some computers unexpectedly power up on installation of a board.

Installation

Note: to install Sapera LT and the X64-CL device driver, logon to the workstation as an administrator or with an account that has administrator privileges.

The Sapera LT Development Library (or 'runtime library' if application development is not being performed) must be installed before the X64-CL device driver.

- Turn the computer off, disconnect the power cord (disables power standby mode), and open the computer chassis to allow access to the expansion slot area.
- Install the X64-CL into a free 64 bit PCI expansion slot. If no 64 bit PCI slot is available, use a common 32 bit PCI slot. X64-CL supports the plug and play automatic configuration of the PCI specification.
- Close the computer chassis and turn the computer on.
- Windows will find the X64-CL and start its **Found New Hardware Wizard**. Click on the **Cancel** button to close the Wizard.
- If using **Windows Vista or Windows 7**, Windows will display its **Found New Hardware dialog**. Click on the default "*Ask me again later*" and continue with the installation. Note that if you select the third option "*Don't show this message again for this device*", there will be no prompt if the Teledyne DALSA board is installed in the same computer.

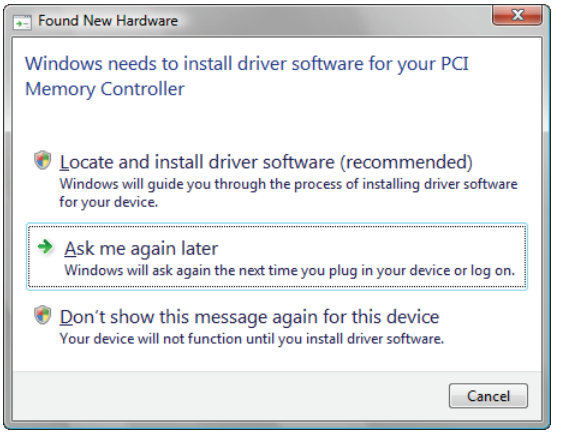

Found New Hardware
Windows needs to install driver software for your PCI Memory Controller
Locate and install driver software (recommended)
Windows will guide you through the process of installing driver software for your device.
Ask me again later
Windows will ask again the next time you plug in your device or log on.
Don't show this message again for this device
Your device will not function until you install driver software.
Cancel

Figure 1: Found New Hardware prompt

Sapera LT Library Installation

- Insert the Teledyne DALSA Sapera CD-ROM. If **AUTORUN** is enabled on your computer, the installation menu is presented.
- If **AUTORUN** is not enabled, use Windows Explorer and browse to the root directory of the CD-ROM. Execute **launch.exe** to start the installation menu and install the required Sapera components.
- The installation program will prompt you to reboot the computer.

Refer to *Sapera LT User's Manual* for additional details about Sapera LT.

X64-CL Driver Installation

The X64-CL board driver supports installation in a Windows XP, Windows Vista, or Windows 7 system.

- After installing Sapera, continue by selecting the X64-CL driver installation.
- If Sapera was previously installed, insert the Teledyne DALSA Sapera Essential CD-ROM to install the board driver. With **AUTORUN** enabled, the installation menu automatically displays. Install the X64-CL driver.
- With **AUTORUN** not enabled, use Windows Explorer and browse to the root directory of the CD-ROM. Execute **launch.exe** to start the installation menu and install the X64-CL driver. During the late stages of the installation, the X64-CL firmware loader application starts. See the description in the following section.
- If Windows displays any unexpected message concerning the installed board, power off the system and verify that the X64-CL is installed properly in the computer slot.

X64-CL Firmware Loader

After Windows boots, the Device Manager-Firmware Loader program automatically executes at the end of the driver installation and on every subsequent reboot of the computer. It will determine if the X64-CL requires a firmware update. If firmware is required, a dialog displays. This dialog also allows the user to load firmware for alternate operational modes of the X64-CL.

***Important*:** In the very rare case of firmware loader errors please see "Recovering from a Firmware Update Error" on page 26.

Firmware Update: Automatic Mode

Click **Automatic** to update the X64-CL firmware. Standard firmware provides the default functionality supported by the X64-CL hardware. X64-CL also supports specialized capabilities (such as non-standard cameras) therefore corresponding firmware is loaded.

See "User Programmable Configurations" on page 7 and for details on all supported modes, selected via a manual firmware update.

If there are multiple X64-CL boards in the system, all boards will be updated with new firmware. If any installed X64-CL boards installed in a system already have the correct firmware version, an update is not required. The figure below shows that the installed X64-CL requires a firmware update.

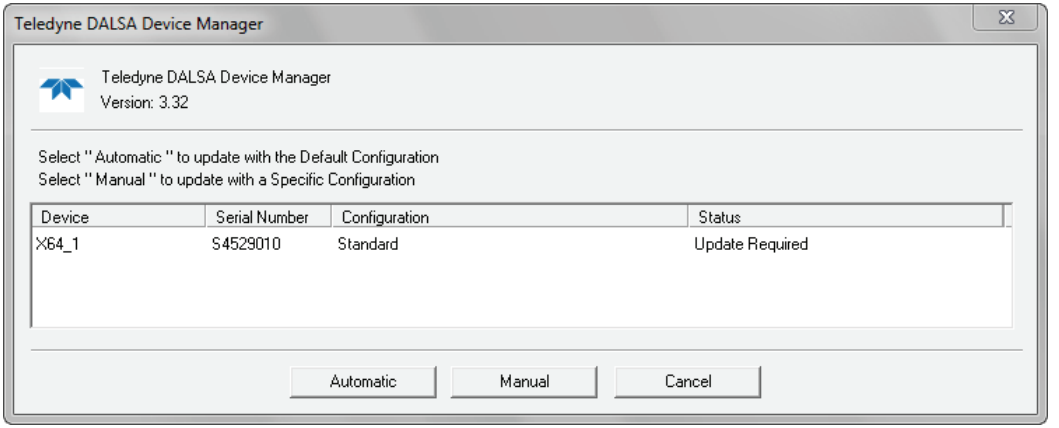


Figure 2: Firmware Update Required

Note: if you are certain that the X64-CL firmware is of the same version as the driver being installed, you can click on **Cancel** to bypass the update procedure.

Firmware Update: Manual Mode

Select **Manual** mode to load firmware other then the default version or when, in the case of multiple X64-CL in the same system, each board requires different firmware.

The figure below shows the Device Manager manual firmware screen. Information on all installed X64-CL boards, their serial numbers, and their firmware components are shown.

A manual firmware update is as follows:

- Select the X64-CL via the selection box (if there are multiple boards in the system)
- From the Configuration field drop menu select the firmware version required
- Click on the Start Update button
- Observe the firmware update progress in the message output window
- Close the Device manager program when the X64-CL board reset complete message is shown

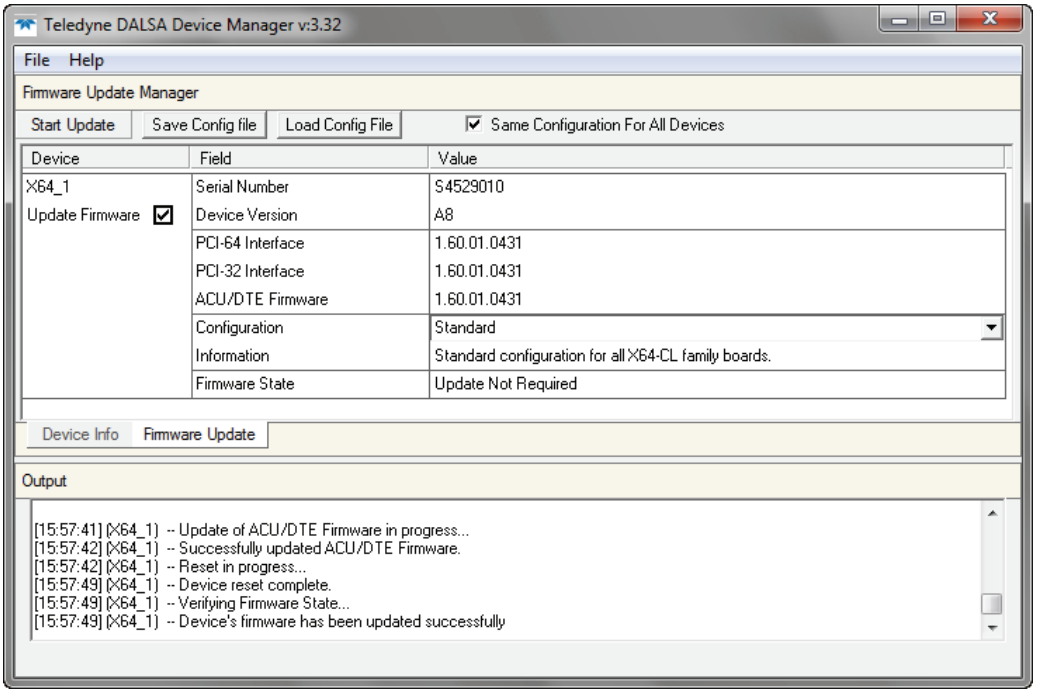


Figure 3: Firmware Selection and Update

Certain non-standard cameras may require custom X64-CL firmware designed to support the enhanced specifications of those cameras. Teledyne DALSA, in the future, may release such non-standard firmware.

Executing the Firmware Loader from the Start Menu

If required, the X64-CL Firmware Loader program is executed via the Windows Start Menu shortcut **Start • All Programs • Teledyne DALSA • X64-CL Device Driver •Firmware Update**.

Important: Not all firmware versions are supported by each version or board revision of X64-CL or X64-CL EM. Contact Teledyne DALSA sales for available X64-CL versions.

Requirements for a Silent Install

Both Sapera LT and the X64-CL driver installations share the same installer technology. When the installations of Teledyne DALSA products are embedded within a third party's product installation, the mode can either have user interaction or be completely silent. The following installation mode descriptions apply to both Sapera and the hardware driver.

Note: You must reboot after the installation of Sapera LT. However, to streamline the installation process, Sapera LT can be installed without rebooting before installing the board hardware device drivers. The installations then completes with a single final system reboot.

Perform Teledyne DALSA embedded installations in either of these two ways:

- **Normal Mode**
  The default mode is interactive. This is identical to running the setup.exe program manually from Windows (either run from Windows Explorer or the Windows command line).
- **Silent Mode**
  This mode requires no user interaction. A preconfigured "response" file provides the user input. The installer displays nothing.

Silent Mode Installation

A Silent Mode installation is recommended when integrating Teledyne DALSA products into your software installation. The silent installation mode allows the device driver installation to proceed without the need for mouse clicks or other input from a user.

Preparing a Silent Mode Installation requires two steps:

- Prepare the response file, which emulates a user.
- Invoke the device driver installer with command options to use the prepared response file.

Creating a Response File

Create the installer response file by performing a device driver installation with a command line switch "-r". The response file is automatically named **`setup.iss`** and is saved in the \windows folder. If a specific directory is desired, the switch –f1 is used.

As an example, to save a response file in the same directory as the installation executable of the X64-CL, the command line would be:

```
X64-CL_1.00.00.0000 -r -f1".\setup.iss"
```

Running a Silent Mode Installation

A device driver silent installation, whether done alone or within a larger software installation requires the device driver executable and the generated response file **`setup.iss`**.

Execute the device driver installer with the following command line:

```
X64-CL_1.00.00.0000 -s -f1".\setup.iss"
```

Where the **–s** switch specifies the silent mode and the **–f1** switch specifies the location of the response file. In this example, the switch –f1".\setup.iss" specifies that the **`setup.iss`** file be in the same folder as the device driver installer.

Note: On Windows Vista and 7, the Windows Security dialog box will appear unless one has already notified Windows to 'Always trust software from "DALSA Corp." during a previous installation of a driver.

Silent Mode Uninstall

Similar to a silent installation, a response file must be prepared first as follows.

Creating a Response File

The installer response file is created by performing a device driver un-installation with a command line switch "-r". The response file is automatically named **`setup_uninstall.iss`** which is saved in the \windows folder. If a specific directory is desired, the switch "–f1" is used.

As an example, to save a response file in the same directory as the installation executable of the X64-CL, the command line would be:

```
X64-CL_1.00.00.0000 -r -f1".\setup_uninstall.iss"
```

Running a Silent Mode Uninstall

Similar to the device driver silent mode installation, the un-installation requires the device driver executable and the generated response file **`setup.iss`**.

Execute the device driver installer with the following command line:

```
X64-CL_1.00.00.0000 -s -f1".\setup_uninstall.iss"
```

Where the **–s** switch specifies the silent mode and the **–f1** switch specifies the location of the response file. In this example, the switch –f1".\setup_uninstall.iss" specifies that the **`setup_uninstall.iss`** file be in the same folder as the device driver installer.

Silent Mode Installation Return Code

A silent mode installation creates a file "corinstall.ini" in the Windows directory. A section called [SetupResult] contains the 'status' of the installation. A value of 1 indicates that the installation has started and a value of 2 indicates that the installation has terminated.

A silent mode installation also creates a log file "setup.log" which by default is created in the same directory and with the same name (except for the extension) as the response file. The /f2 option enables you to specify an alternative log file location and file name, as in Setup.exe /s /f2"C:\Setup.log".

The "setup.log" file contains three sections. The first section, [InstallShield Silent], identifies the version of InstallShield used in the silent installation. It also identifies the file as a log file. The second section, [Application], identifies the installed application name, version, and the company name. The third section, [ResponseResult], contains the 'ResultCode' indicating whether the silent installation succeeded. A value of 0 means the installation was successful.

Installation Setup with CorAppLauncher.exe

The installation setup can be run with the CorAppLauncher.exe tool provided with the driver.

- Install the board driver and get CorAppLauncher.exe from the \bin directory of the installation.
- When running the installation, CorAppLauncher.exe will return only when the installation is finished.
- When run from within a batch file, obtain the installation exit code from the ERRORLEVEL value.
- The arguments to CorAppLauncher.exe are
  -l: Launch application
  -f: Application to launch. Specify a fully qualified path.

As an example:

- CorAppLauncher –l –f"c:\driver_install\x64-cl_1.00.00.0000.exe"
- IF %ERRORLEVEL% NEQ 0 goto launch error

Note: There is a 32-bit and 64-bit version of CorAppLauncher.exe. When installing the driver, only the version related to the OS is installed. However, the 32-bit version is usable on either 32-bit or 64-bit Windows.

Custom Driver Installation using install.ini

Customize the driver installation by parameters defined in the file “install.ini”. By using this file, the user can:

- Select the user default configuration.
- Select different configurations for systems with multiple boards.
- Assign a standard Serial COM port to board.

Creating the install.ini File

- Install the driver in the target computer. All X64-CL boards required in the system must be installed.
- Configure each board’s acquisition firmware using the Teledyne DALSA Device Manager tool (see Firmware Update: Manual Mode).
- If a standard Serial COM port is required for any board, use the Sapera Configuration tool (see Using the Camera Link Serial Control Port).
- When each board setup is complete, using the Teledyne DALSA Device Manager tool, click on the Save Config File button. This will create the “install.ini” file.

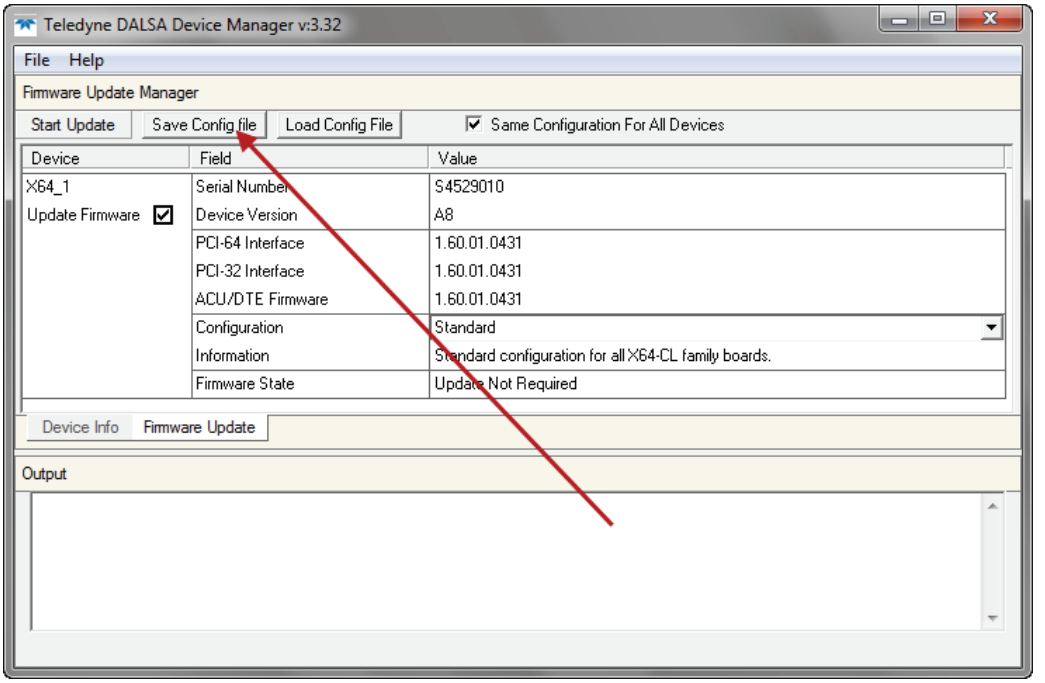


Figure 4: Create an “install.ini” File

Run the Installation using install.ini

Copy the install.ini file into the same directory as the setup installation file. Run the setup installation as normal. The installation will automatically check for an install.ini file and if found, use the configuration defined in it.

Upgrading Sapera or Board Driver

When installing a new version of Sapera or an acquisition board driver in a computer with a previous installation, the current version **must** be un-installed first. Upgrade scenarios are described below.

Board Driver Upgrade Only

Minor upgrades to acquisition board drivers are typically distributed as ZIP files available in the Teledyne DALSA web site http://www.teledynedalsa.com/mv/support. Board driver revisions are also available on the next release of the Sapera CD-ROM.

Often minor board driver upgrades do not require a new revision of Sapera. To confirm that the current Sapera version will work with the new board driver:

- Check the new board driver ReadMe.txt file before installing, for information on the minimum Sapera version required.
- If the ReadMe.txt file does not specify the Sapera version, you can contact Technical Support (see "Technical Support" on page 114).

To upgrade the board driver only:

- Logon the computer as an administrator or with an account that has administrator privileges.
- In **Windows XP**, from the start menu select **Start • Settings • Control Panel • Add or Remove Programs**. Select the Teledyne DALSA X64-CL board driver and click **Remove**.
- In **Windows Vista/7**, from the start menu select **Start • Settings • Control Panel • Programs and Features**. Double-click the Teledyne DALSA X64-CL board driver and click **Remove**.
    - **Windows XP only:**
        - When the driver un-install is complete, reboot the computer.
        - Logon the computer as an administrator again.
- Install the new board driver. Run **Setup.exe** if installing manually from a downloaded driver file.
- If the new driver is on a Sapera CD-ROM follow the installation procedure described in "X64-CL Driver Installation" on page 12.
- Note that you cannot install a Teledyne DALSA board driver without Sapera LT installed on the computer.

Upgrading both Sapera and Board Driver

When both Sapera and the Teledyne DALSA acquisition board driver are upgraded, follow the procedure described below.

- Logon the computer as an administrator or with an account that has administrator privileges.
- In **Windows XP**, from the start menu select **Start • Settings • Control Panel • Add or Remove Programs**. Select the Teledyne DALSA X64-CL board driver and click **Remove**. Follow by also removing the older version of Sapera LT.
- In **Windows Vista/7**, from the start menu select **Start • Settings • Control Panel • Programs and Features**. Double-click the Teledyne DALSA X64-CL board driver and click **Remove**. Follow by also removing the older version of Sapera LT.
- Reboot the computer and logon the computer as an administrator again.
- Install the new versions of Sapera and the board driver as if this was a first time installation. See "Installation" on page 11 and "X64-CL Driver Installation" on page 12 for installation procedures.

Using the Camera Link Serial Control Port

The Camera Link cabling specification includes a serial communication port for direct camera control by the frame grabber (see "J1: Camera Link Connector 1 (applies to X64-CL Dual & Full models)" on page 87). The X64-CL driver supports this serial communication port either directly or by mapping it to a host computer COM port. Any serial port communication program, such as Windows HyperTerminal, can connect to the camera in use and modify its function modes via its serial port controls. The X64-CL serial port supports communication speeds from 9600 up to 115 kbps.

Note: if your serial communication program can directly select the X64-CL serial port then mapping the X64-CL serial port to a system COM port is not necessary.

The X64-CL serial port is mapped to an available COM port by using the Sapera Configuration tool. Run the program from the Windows start menu: **Start •All Programs • Teledyne DALSA • Sapera LT • Sapera Configuration**.

COM Port Assignment

The lower section of the Sapera Configuration program screen contains the serial port configuration menu. Configure as follows:

- Use the **Physical Port** drop menu to select the Sapera board device from all available Sapera boards with serial ports (when more then one board is in the system).
- Use the **Maps to** drop menu to assign an available COM number to that Sapera board serial port.
- Click on the **Save Settings Now** button then the **Close** button. You are prompted to reboot your computer to enable the serial port mapping.
- The X64-CL serial port, now mapped to COM3 in this example, is available as a serial port to any serial port application for camera control. Note that this serial port is not listed in the **Windows Control Panel•System Properties•Device Manager** because it is a logical serial port mapping.
- An example setup using Windows HyperTerminal follows (see "Setup Example with Windows HyperTerminal" on page 22).

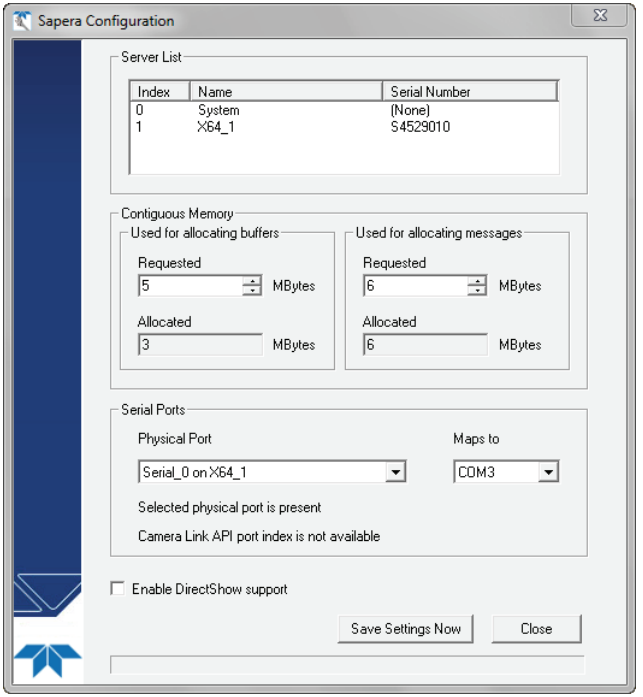


Figure 5: Sapera Configuration Program

Setup Example with Windows HyperTerminal

- Run HyperTerminal and type a name for the new connection when prompted. Then click OK.
- On the following dialog screen select the COM port to connect with. In this example the X64-CL serial port was previously mapped to COM3 by the Sapera Configuration program.
- Note that HyperTerminal is not available in Windows Vista or Windows 7, but is available for download from various Internet locations.

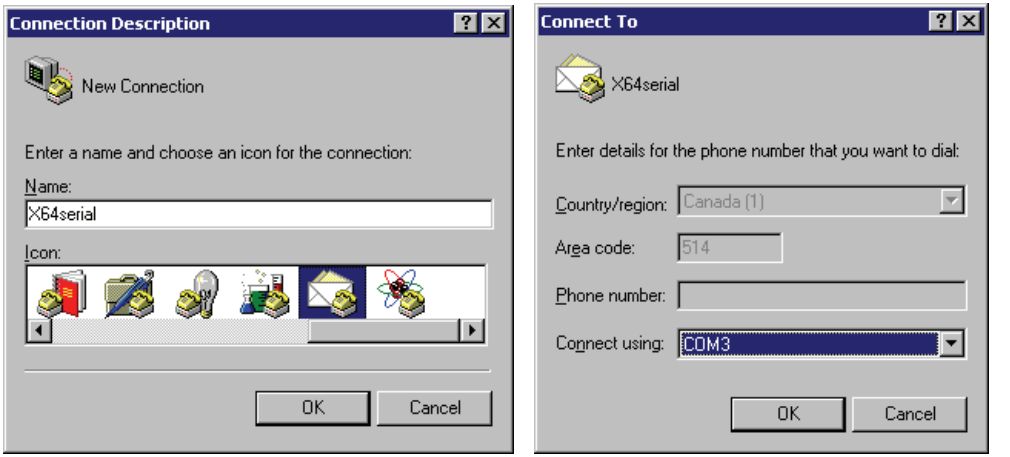
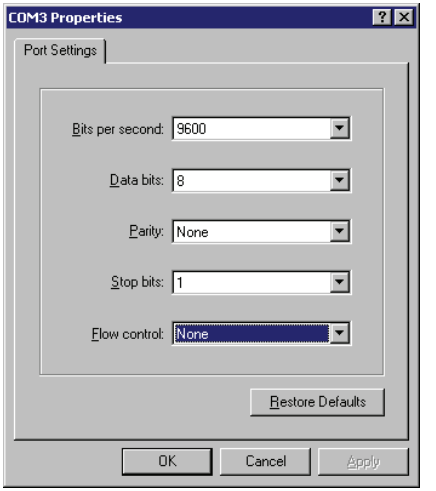


Figure 6: Hyperterminal setup

- HyperTerminal now presents a dialog to configure the COM port properties. Change settings as required by the camera you are connecting to. Note that the X64-CL serial port does not support hardware flow control.



Displaying X64-CL Board Information

The Device Manager program also displays information about the X64-CL boards installed in the system. To view board information run the program via the Windows Start Menu shortcut **Start •All Programs • Teledyne DALSA • X64-CL Device Driver • Device Manager**.

Device Manager – Board Viewer

The following screen image shows the Device Manager program with the Information tab active. The left window displays all X64-CL boards in the system and their individual device components. The right window displays the information stored in the selected board device. This example screen shows two X64-CL boards installed and the information displayed for the EEProm on the first X64-CL board.

The X64-CL report file (BoardInfo.txt) is generated by clicking **File • Save Device Info**. This report file may be requested by Teledyne DALSA Technical Support to aid in troubleshooting installation or operational problems.

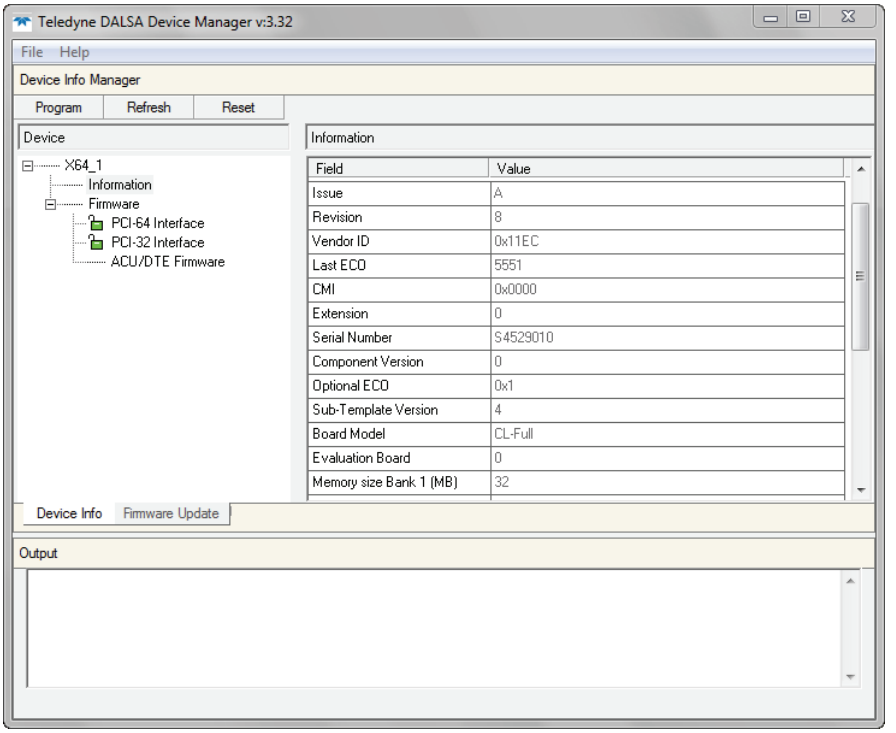


Figure 7: Getting Board Information

Configuring Sapera

Viewing Installed Sapera Servers

The Sapera configuration program (**Start •All Programs • Teledyne DALSA • Sapera LT • Sapera Configuration**) allows the user to see all available Sapera servers for the installed Sapera-compatible boards.

The **System** entry represents the system server. It corresponds to the host machine (your computer) and is the only server that should always be present. As shown in the following screen image, server index 1 is the installed X64-CL board.

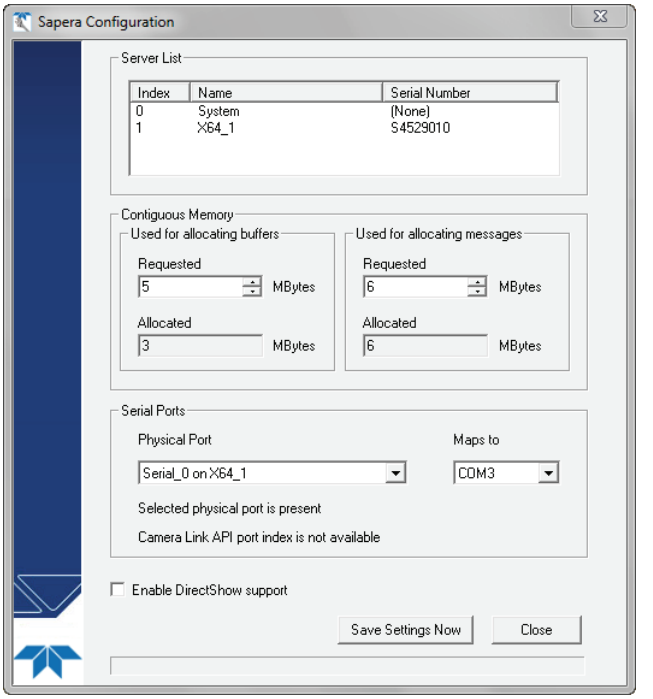


Figure 8: Sapera Configuration Program

Increasing Contiguous Memory for Sapera Resources

The **Contiguous Memory** section lets the user specify the total amount of contiguous memory (a block of physical memory, occupying consecutive addresses) reserved for the resources needed for **Sapera buffers** allocation and **Sapera messaging**. For both items, the **Requested** value dialog box shows the driver default memory setting while the **Allocated** value displays the amount of contiguous memory that has been allocated successfully. The default values will generally satisfy the needs of most applications.

The **Sapera buffers** value determines the total amount of contiguous memory reserved at boot time for the allocation of dynamic resources used for frame buffer management such as scatter-gather list, DMA descriptor tables plus other kernel needs. Adjust this value higher if your application generates any out-of-memory error while allocating host frame buffers or when connecting the buffers via a transfer object. You can approximate the amount of contiguous memory required as follows:

- Calculate the total amount of host memory used for frame buffers
  [number of frame buffers • number of pixels per line • number of lines • (2 - if buffer is 10 or 12 bits)].
- Provide 1MB for every 256 MB of host frame buffer memory required.
- Add an additional 1 MB if the frame buffers have a short line length, say 1k or less
  (the increased number of individual frame buffers requires more resources).
- Add an additional 2 MB for various static and dynamic Sapera resources.
- Add the amount of memory needed for DMA tables using the formula (Sapera 7.10 and up):
  [number of frame buffers • number of lines • 16 • (line length in bytes / 4kB)].
- Test for any memory error when allocating host buffers. Simply use the Buffer menu of the Sapera Grab demo program (see "Using the Grab Demo" on page 49) to allocate the number of host buffers required for your acquisition source. Feel free to test the maximum limit of host buffers possible on your host system – the Sapera Grab demo will not crash when the requested number of host frame buffers cannot be allocated.

Host Computer Frame Buffer Memory Limitations

When planning a Sapera application and its host frame buffers used, plus other Sapera memory resources, do not forget the Windows operating system memory needs.

A Sapera application using the preferred *scatter gather buffers* could consume most of the remaining system memory, with a large allocation of frame buffers. If using frame buffers allocated as a *single contiguous memory block*, Windows will limit the allocation dependent on the installed system memory. Use the Buffer menu of the Sapera Grab demo program to allocate host buffer memory until an error message signals the limit allowed by the operating system used.

Contiguous Memory for Sapera Messaging

The current value for **Sapera messaging** determines the total amount of contiguous memory reserved at boot time for messages allocation. This memory space is used to store arguments when a Sapera function is called. Increase this value if you are using functions with large arguments, such as arrays and experience any memory errors.

Troubleshooting Installation Problems

The X64-CL (and the X64 family of products) has been tested by Teledyne DALSA in a wide variety of 64-bit and 32-bit PCI computers. Although unlikely, installation problems may occur due to the constant changing nature of computer equipment and operating systems. This section describes what the user can verify to determine the problem or the checks to make before contacting Technical Support. Note that information provided within this section will be updated with the latest information Teledyne DALSA can provide for each manual version released.

If you require help and need to contact Technical Support, make detailed notes on your installation and/or test results for our technical support to review. See “Technical Support” on page 114 for contact information.

Recovering from a Firmware Update Error

This procedure is required if any failure occurred while updating the X64-CL firmware on installation or during a manual firmware upgrade. On the rare occasion the board has corrupted firmware, any Sapera application such as CamExpert or the grab demo program will not find an installed board to control.

Possible reasons for firmware loading errors or corruption are:

- Computer system mains power failure or deep brown-out.
- PCI bus or checksum errors.
- PCI bus timeout conditions due to other devices.
- User forcing a partial firmware upload using an invalid firmware source file.

When the X64-CL firmware is corrupted, executing a manual firmware upload will not work because the firmware loader can not communicate with the board. In an extreme case, corrupted firmware may even prevent Windows from booting.

***Solution*:** The user manually forces the board to initialize from protected firmware designed only to allow driver firmware uploads. When the firmware upload is complete, the board is then rebooted to initialize in its normal operational mode.

- This procedure requires removing the X64-CL board several times from the computer.
- *Important*: Referring to the board's user manual (in the connectors and jumpers reference section), identify the configuration jumper location. The Boot Recovery Mode jumper for the X64-CL is J11 or J14 for the X64-CL EM (see "J11 (CL), J14 (CL EM): Start Mode" on page 106).
- Shut down Windows and power OFF the computer.
- Move the configuration switch for boot recovery (safe mode) from its default position to the boot recovery mode position.
- Power on the computer. Windows will boot normally.
- When Windows has started, do a manual firmware update procedure to update the firmware again (see "Executing the Firmware Loader from the Start Menu" on page 15).
- When the update is complete, shut down Windows and power off the computer.

- Set the Boot Recovery Mode switch back to its default position and reboot the computer once again.
- Verify that the frame grabber is functioning by running a Sapera application such as CamExpert. The Sapera application will now be able to communicate with the X64-CL board.

Windows Event Viewer

Windows Event Viewer (**Computer Management • System Tools • Event Viewer**), lists various events that have taken place during the Operating System boot sequence. If a driver generates an error, it will normally log an entry in the event list.

Teledyne DALSA Device Manager Program

The Device Manager program provides a convenient method of collecting information about the installed X64-CL. System information such as operating system, computer CPU, system memory, PCI configuration space, plus X64-CL firmware information can be displayed or written to a text file (default file name – BoardInfo.txt). Note that this is a second function mode of the same program used to manually upload firmware to the X64-CL.

Execute the program via the Windows Start Menu shortcut **Start • All Programs • Teledyne DALSA • X64 Device Driver • Device Manager**. If the Device Manager program does not run, it will exit with a message that the board was not found. Since the X64-CL board must have been in the system to install the board driver, possible reasons for an error are:

- Board was removed
- Board driver did not start or was terminated
- PCI conflict after some other device was installed

Information Window

The following figure shows the Device Manager information screen. Click to highlight one of the board components and the information for that item is shown on the right hand window.

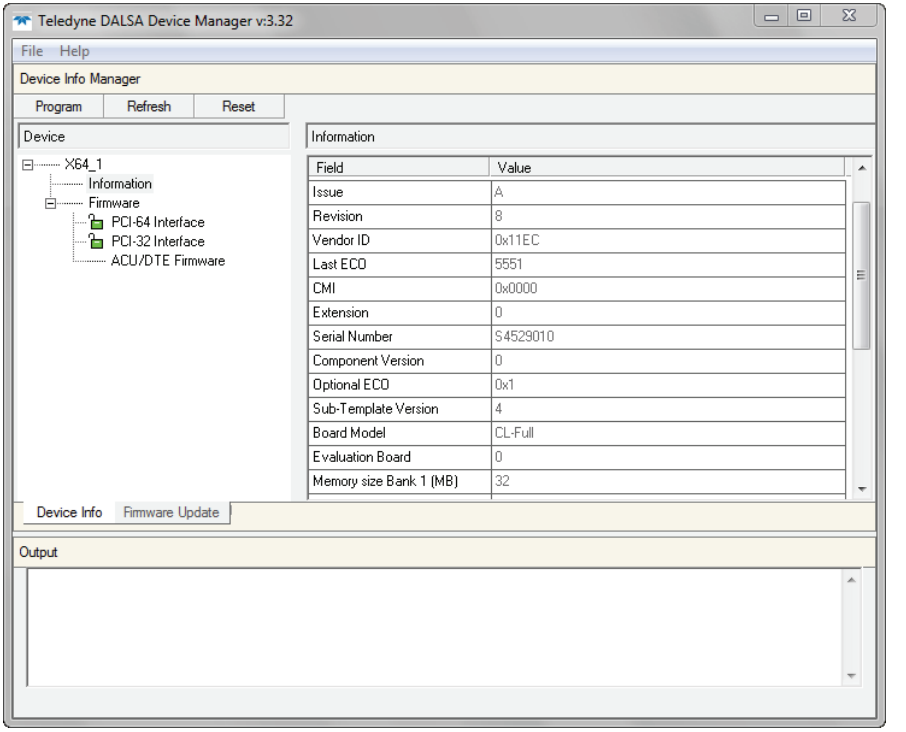


Figure 9: Device Manager Information Screen

- Click on **File • Save Device Info** to save all information to a text file. Default location is *drive*:\Program Files\Teledyne DALSA\X64-CL\Bin\BoardInfo.txt. Email this file when requested by Technical Support.

PCI Configuration

One of the first items to check when there is a problem with any PCI board is to examine the system PCI configuration and ensure that there are no conflicts with other PCI or system devices. The *Teledyne DALSA PCI Diagnostic* program (**cpcidiag.exe**) allows examination of the PCI configuration registers and can save this information to a text file. Run the program via the Windows Start Menu shortcut **Start • All Programs • Teledyne DALSA • Sapera LT • Tools • PCI Diagnostics**.

As shown in the following screen image, use the first drop menu to select the PCI device to examine. Select the device "X64-CL from Teledyne DALSA". Note the bus and slot number of the installed board (this will be unique for each system unless systems are setup identically). Click on the **Diagnostic** button to view an analysis of the system PCI configuration space.

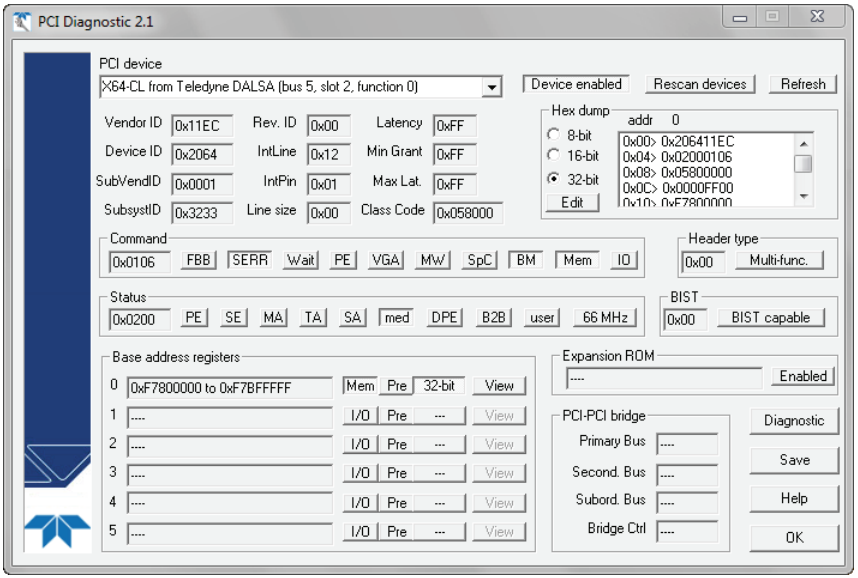


Figure 10: PCI Diagnostic program main menu

Clicking on the **Diagnostic** button opens a new window with the diagnostic report. From the PCI Bus Number drop menu select the bus number that the X64-CL is installed in. In this example the X64-CL is on bus 5.

The window now shows the I/O and memory ranges used by each device on the selected PCI bus. The information display box will detail any PCI conflicts. If there is a problem, click on the **Save** button. A file named '**pcidiag.txt'** is created (in the ..\Sapera\bin directory) with a full dump of the PCI configuration registers. Email this file when requested by the Teledyne DALSA Technical Support group along with a full description of your computer.

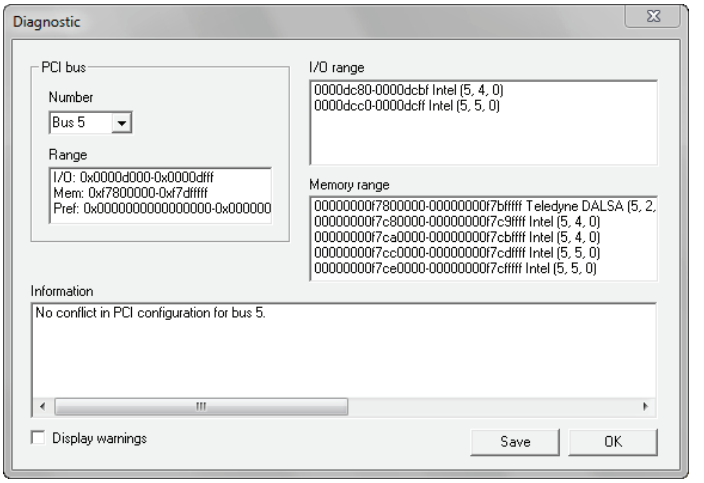


Figure 11: PCI Diagnostic program diagnostic menu

Sapera and Hardware Windows Drivers

The next step is to make certain the appropriate Teledyne DALSA drivers have started successfully during the boot sequence. Example, in Windows 2000 click on the **Start • All Programs • Accessories • System Tools • System Information • Software Environment**. Click on **System Drivers** (Windows XP). Make certain the following drivers have started for the X64-CL driver.

Device	Description	Type	Started
CorX64	X64-CL messaging	Kernel Driver	Yes
CorLog	Sapera Log viewer	Kernel Driver	Yes
CorMem	Sapera Memory manager	Kernel Driver	Yes
CorPci	Sapera PCI configuration	Kernel Driver	Yes
CorSerial	Sapera Serial Port manager	Kernel Driver	Yes

Table 1: X64-CL Device Drivers

Teledyne DALSA Technical Support may request that you check the status of these drivers as part of the troubleshooting process.

Log Viewer

The third step in the verification process is to save in a text file the information collected by the Log Viewer program. Run the program via the Windows Start Menu shortcut **Start • All Programs • Teledyne DALSA • Sapera LT • Tools • Log Viewer**.

The Log Viewer lists information about the installed Teledyne DALSA drivers. Click on File • Save and you will be prompted for a text file name to save the Log Viewer contents. Email this text file to Technical Support when requested or as part of your initial contact email.

Although the information collected by the Log Viewer seems complicated, you can make some initial diagnostics by checking the status of the Teledyne DALSA driver. In the screen shot below, note the highlighted line which states [... CORX64L.DLL ... Found 1 X64 board (s) ...]. This confirms that the driver can communicate with the X64-CL.

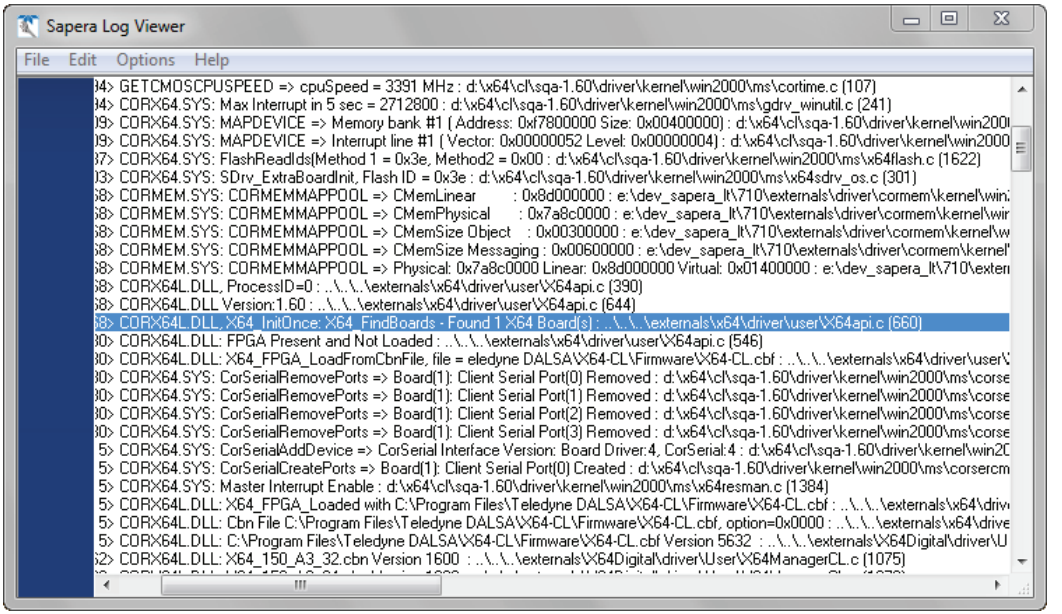

Sapera Log Viewer
File Edit Options Help
)4> GETCMOSCPUSPEED => cpuSpeed = 3391 MHz : d:\x64\cl\sqa-1.60\driver\kernel\win2000\ms\cortime.c (107)
)4> CORX64.SYS: Max Interrupt in 5 sec = 2712800 : d:\x64\cl\sqa-1.60\driver\kernel\win2000\ms\gdrv_winutil.c (241)
)9> CORX64.SYS: MAPDEVICE => Memory bank #1 (Address: 0xf7800000 Size: 0x00400000) : d:\x64\cl\sqa-1.60\driver\kernel\win200
)9> CORX64.SYS: MAPDEVICE => Interrupt line #1 (Vector: 0x00000052 Level: 0x00000004) : d:\x64\cl\sqa-1.60\driver\kernel\win2000
)7> CORX64.SYS: FlashReadIds(Method 1 = 0x3e, Method2 = 0x00 : d:\x64\cl\sqa-1.60\driver\kernel\win2000\ms\x64flash.c (1622)
)3> CORX64.SYS: SDrv_ExtraBoardInit, Flash ID = 0x3e : d:\x64\cl\sqa-1.60\driver\kernel\win2000\ms\x64sdrv_os.c (301)
)8> CORMEM.SYS: CORMEMMAPPOOL => CMemLinear : 0x8d000000 : e:\dev_sapera_lt\710\externals\driver\cormem\kernel\win
)8> CORMEM.SYS: CORMEMMAPPOOL => CMemPhysical : 0x7a8c0000 : e:\dev_sapera_lt\710\externals\driver\cormem\kernel\wir
)8> CORMEM.SYS: CORMEMMAPPOOL => CMemSize Object : 0x00300000 : e:\dev_sapera_lt\710\externals\driver\cormem\kernel\w
)8> CORMEM.SYS: CORMEMMAPPOOL => CMemSize Messaging : 0x00600000 : e:\dev_sapera_lt\710\externals\driver\cormem\kernel'
)8> CORMEM.SYS: CORMEMMAPPOOL => Physical: 0x7a8c0000 Linear: 0x8d000000 Virtual: 0x01400000 : e:\dev_sapera_lt\710\exter
)8> CORX64L.DLL, ProcessID=0 : ..\..\..\externals\x64\driver\user\X64api.c (390)
)8> CORX64L.DLL Version:1.60 : ..\..\..\externals\x64\driver\user\X64api.c (644)
)8> CORX64L.DLL, X64_InitOnce: X64_FindBoards - Found 1 X64 Board(s) : ..\..\..\externals\x64\driver\user\X64api.c (660)
)0> CORX64L.DLL: FPGA Present and Not Loaded : ..\..\..\externals\x64\driver\user\X64api.c (546)
)0> CORX64L.DLL: X64_FPGA_LoadFromCbnFile, file = eledyne DALSA\X64-CL\Firmware\X64-CL.cbf : ..\..\..\externals\x64\driver\user\
)0> CORX64.SYS: CorSerialRemovePorts => Board(1): Client Serial Port(0) Removed : d:\x64\cl\sqa-1.60\driver\kernel\win2000\ms\corse
)0> CORX64.SYS: CorSerialRemovePorts => Board(1): Client Serial Port(1) Removed : d:\x64\cl\sqa-1.60\driver\kernel\win2000\ms\corse
)0> CORX64.SYS: CorSerialRemovePorts => Board(1): Client Serial Port(2) Removed : d:\x64\cl\sqa-1.60\driver\kernel\win2000\ms\corse
)0> CORX64.SYS: CorSerialRemovePorts => Board(1): Client Serial Port(3) Removed : d:\x64\cl\sqa-1.60\driver\kernel\win2000\ms\corse
5> CORX64.SYS: CorSerialAddDevice => CorSerial Interface Version: Board Driver:4, CorSerial:4 : d:\x64\cl\sqa-1.60\driver\kernel\win2C
5> CORX64.SYS: CorSerialCreatePorts => Board(1): Client Serial Port(0) Created : d:\x64\cl\sqa-1.60\driver\kernel\win2000\ms\corsercm
5> CORX64.SYS: Master Interrupt Enable : d:\x64\cl\sqa-1.60\driver\kernel\win2000\ms\x64resman.c (1384)
5> CORX64L.DLL: X64_FPGA_Loaded with C:\Program Files\Teledyne DALSA\X64-CL\Firmware\X64-CL.cbf : ..\..\..\externals\x64\driv
5> CORX64L.DLL: Cbn File C:\Program Files\Teledyne DALSA\X64-CL\Firmware\X64-CL.cbf, option=0x0000 : ..\..\..\externals\x64\drive
5> CORX64L.DLL: C:\Program Files\Teledyne DALSA\X64-CL\Firmware\X64-CL.cbf Version 5632 : ..\..\..\externals\X64Digital\driver\U
52> CORX64L.DLL: X64_150_A3_32.cbn Version 1600 : ..\..\..\externals\X64Digital\driver\User\X64ManagerCL.c (1075)

Figure 12: Sapera Log Viewer output

Windows Device Manager

An alternative method to confirm the installation of the X64-CL board and driver is to use the Windows Device manager tool. Use the Start Menu shortcut **Start • … • Control Panel • System • Hardware • Device Manager**. As shown in the following screen images, look for X64 board under "Imaging Devices" (screen image shows two X64-CL boards installed). Double-click and look at the device status. You should see "This device is working properly." Go to "Resources" tab and make certain that the device is mapped and has an interrupt assigned to it, without any conflicts.

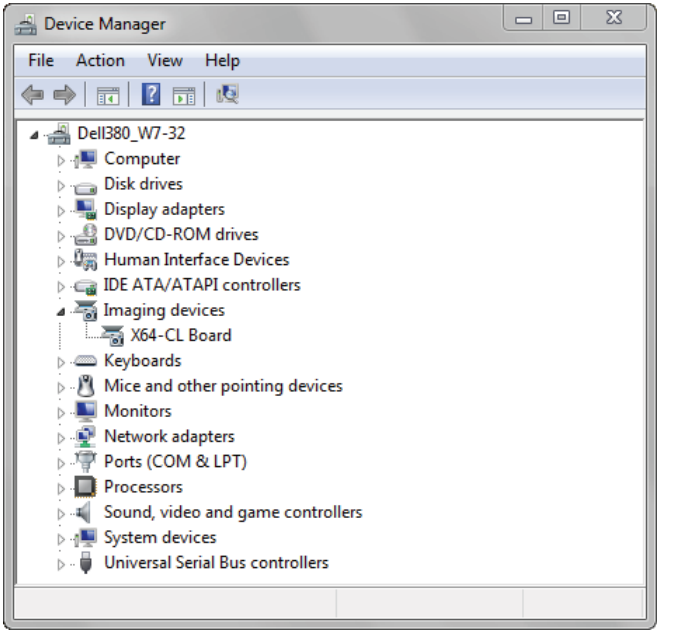


Figure 13: Windows Device Manager list

Memory Requirements with Area Scan Acquisitions

The X64-CL allocates by default two frame buffers in on-board memory, each equal in size to the acquisition frame buffer. This double buffering memory allocation is automatic at the driver level. Two buffers will ensure that the acquired video frame is complete and not corrupted in cases where the image transfer to host system memory may be interrupted and delayed by other host system processes. That is, there is no interruption to the image acquisition of one buffer by any delays in transfer of the other buffer (which contains the previously acquired video frame) to system memory. Note that an application can change the number of on-board frame buffers using the Sapera LT API.

If allocation for the requested number of buffers fails, the driver will reduce the number of onboard frame buffers requested until they can all fit. When reaching 2 on-board buffers, if they still cannot fit, the driver will reduce the size such that it allocates two partial buffers. This mode will write image data to the buffer while wrapping image lines around to the beginning of a buffer when full. This mode relies on reading out the image data to the host computer faster than the acquisition.

Symptoms: CamExpert Detects no Boards

- **If using Sapera version 5.20 or later:**
  When starting CamExpert, if no Teledyne DALSA board is detected, CamExpert will start in offline mode. There is no error message and CamExpert is functional for creating or modifying a camera configuration file. If CamExpert should have detected the installed board, troubleshoot the installation problem as described below.

Troubleshooting Procedure

When CamExpert detects no installed Teledyne DALSA board, there could be a hardware problem, a PnP problem, a PCI problem, a kernel driver problem, or a software installation problem.

- Make certain that the card is properly seated in PCI slot.
- Perform all installation checks described in this section ("Troubleshooting Installation Problems" on page 26) before contacting Technical Support.
- Try the board in a different PCI slot if it is not seen.

Symptoms: X64-CL Does Not Grab

You are able to start Sapera CamExpert but you do not see an image and the frame rate displayed is 0.

- Verify power is connected to the camera.
- Verify the camera and timing parameters with the camera in free run mode.
- Make certain that you provide an external trigger if the camera configuration file requires one (minimum trigger pulse = 100ns). Use the software trigger feature of CamExpert if you do not have a trigger source.
- Make certain that the camera is properly connected to the cable.
- Make certain that the camera is configured for the proper mode of operation. This must match the camera configuration file. Refer to your camera datasheet.
- Try to snap one frame instead of continuous grab.
- Perform all installation checks described in this section ("Troubleshooting Installation Problems" on page 26) before contacting Technical Support.

Symptoms: Card grabs black

You are able to use Sapera CamExpert, displayed frame rate is as expected, but the display stays black.

- Set your camera to manual exposure mode and set the exposure to a longer period plus open the lens iris.
- Try to snap one frame instead of continuous grab.
- Make certain that the input LUT is not programmed to output all ‘0’s.
- This problem is sometimes caused by a PCI transfer issue. No PCI transfer takes place, so the frame rate is above 0 but nevertheless no image is displayed in CamExpert.
- Make certain that BUS MASTER bit in the PCI configuration space is activated. Look in PCI Diagnostics for **BM** button under “Command” group. Make certain that the **BM** button is activated.
- Perform all installation checks described in this section ("Troubleshooting Installation Problems" on page 26) before contacting Technical Support.

Symptoms: Card acquisition bandwidth is less than expected

The X64-CL is installed in a PCI-64 slot but the acquisition bandwidth is less than expected.

- Review the system for problems or conflicts with other expansion boards or drivers.
- Remove other PCI-32 or PCI-64 boards and check acquisition bandwidth again. Engineering has seen this case where other PCI boards in some systems cause limitations in PCI-64 transfers. Each system, with its combination of system motherboard and PCI boards, will be unique and will need to be tested for bandwidth limitations affecting the imaging application.

CamExpert Quick Start for the X64-CL

Interfacing Cameras with CamExpert

CamExpert is the camera interfacing tool for frame grabber boards supported by the Sapera library. CamExpert generates the Sapera camera configuration file (*yourcamera.ccf*) based on timing and control parameters entered. For backward compatibility with previous versions of Sapera, CamExpert also reads and writes the *.cca and *.cvi camera parameter files.

Every Sapera demo program starts by a dialog window to select a camera configuration file. Even when using the X64-CL with common video signals, a camera file is required. Therefore, CamExpert is typically the first Sapera application run after an installation. Existing .ccf files can be copied to the new installation when similar cameras are used.

CamExpert Example with a Monochrome Camera

The image below shows CamExpert controlling the X64-CL Full. The camera outputs monochrome 8-bit video on a Camera Link interface. After selecting the camera model, the timing parameters are displayed and the user can test by clicking on *Grab*. Descriptions of the CamExpert windows follow the image.

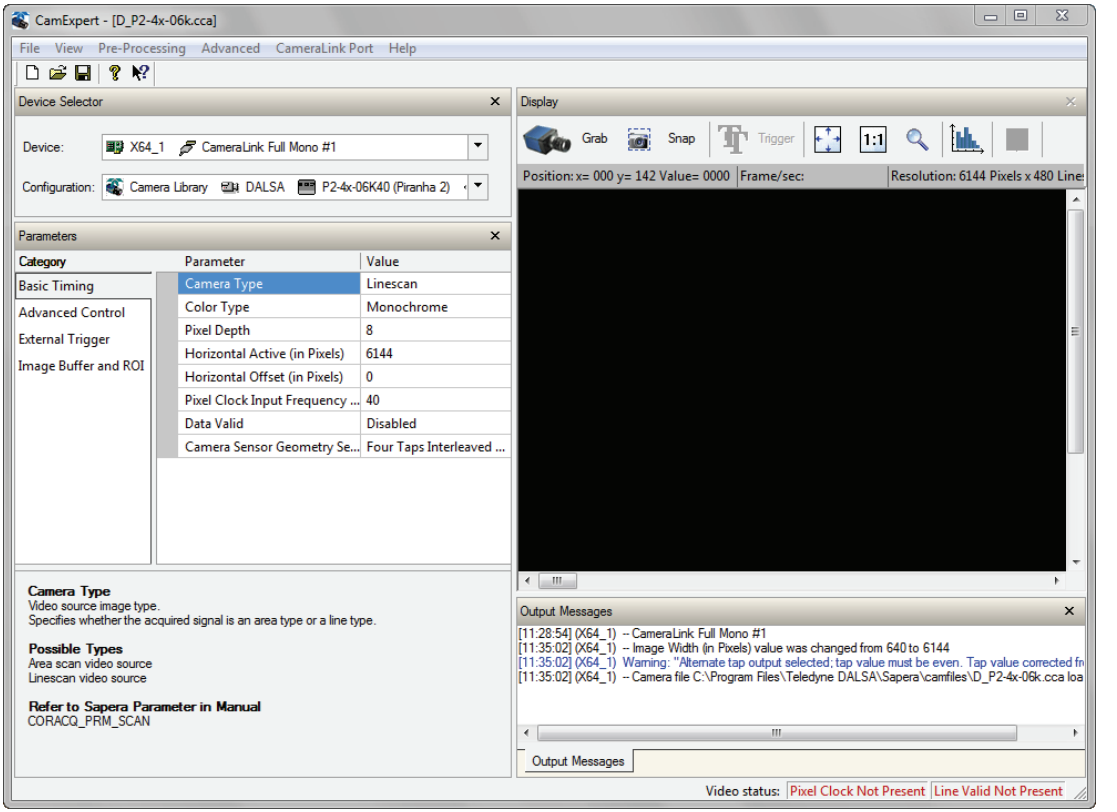


Figure 14: CamExpert Program

CamExpert groups parameters into functional categories. The parameters shown depend on the frame grabber used and what camera is connected. The parameter values are either the camera defaults or the last stored value when the camera was used.

- **Device Selector:** Two drop menus to select which device and which saved configuration to use.
  - **Device:** Select which acquisition device to control and configure a camera file. Required in cases where there are multiple boards in a system and when one board supports multiple acquisition types.
  - **Configuration:** Select the timing for a specific camera model included with the Sapera installation or a standard video standard. The *User's* subsection is where user created camera files are stored.

- **Parameter Groups:** Select a function category and change parameter values as required. Descriptions for the camera parameters change dependent on the camera.
  - **Basic Timing:** Provides or modify static camera parameters.
  - **Advanced Controls:** Select or modify camera specific parameters such as integration method and camera link control assignments.

- **External Trigger:** Parameters to configure the external trigger characteristics.
- **Image Buffer and ROI:** Allows control of the host buffer dimension and format.

- **Display:** An important component of CamExpert is its live acquisition display window, which allows immediate verification of timing or control parameters without the need to run a separate acquisition program. **Grab** starts continuous acquisition (button then toggles to **Freeze** to stop). **Snap** is a single frame grab. **Trigger** is a software trigger to emulate an external source.
- **Output Messages and Video Status Bar:** Events and errors are logged for review. Camera connection status is displayed where green indicates signal present.
- **Camera Link Serial Command:** Select this Tab to open a serial command port to the camera. This allows the user to issue configuration commands if supported by the camera.

For context sensitive help click on the button then click on a camera configuration parameter. A popup provides a short description of the configuration parameter. Click on the button to open the help file for more descriptive information on CamExpert.

CamExpert Demonstration and Test Tools

The CamExpert utility also includes a number of demonstration features which make CamExpert the primary tool to configure, test and calibrate your camera and imaging setup. Display tools include, image pixel value readout, image zoom, and line profiler.

Camera Types & Files Applicable to the X64-CL

The X64-CL supports digital area scan or linescan cameras using the Camera Link interface standard. See "X64-CL Connector View" on page 85 for information on connecting a Camera Link camera to either the X64-CL Full or X64-CL Dual frame grabber.

Contact Teledyne DALSA or browse our web site [http://www.teledynedalsa.com/mv/support] for the latest information and application notes on X64-CL supported cameras.

Camera Files Distributed with Sapera

The Sapera distribution CDROM includes camera files for a selection of X64-CL supported cameras. Using the Sapera CamExpert program, you may use the camera files (CCA) provided to generate a camera configuration file (CCF) that describes the desired camera and frame grabber configuration..

Teledyne DALSA continually updates a camera application library composed of application information and prepared camera files. Along with the camera search utility on the web site, as described above, a number of camera files are ready to download. Camera files are ASCII text and can be read with Windows Notepad on any computer without having Sapera installed.

Overview of Sapera Acquisition Parameter Files (*.ccf or *.cca/*.cvi)

Concepts and Differences between the Parameter Files

There are two components to the legacy Sapera acquisition parameter file set: CCA files (also called cam-files) and CVI files (also called VIC files, i.e. video input conditioning). The files store video-signal parameters (CCA) and video conditioning parameters (CVI), which in turn simplifies programming the frame-grabber acquisition hardware for the camera in use. **Sapera LT 5.0** introduces a new camera configuration file (**CCF**) that combines the CCA and CVI files into one file.

Typically, a camera application will use a CCF file per camera operating mode (or one CCA file in conjunction with several CVI files, where each CVI file defines a specific camera operating mode). An application can also have multiple CCA/CCF files so as to support different image format modes supported by the camera or sensor (such as image binning or variable ROI).

CCF File Details

Files using the ".CCF" extension, (Camera Configuration files), are essentially the camera (CCA) and frame grabber (CVI) parameters grouped into one file for easier configuration file management. This is the default Camera Configuration file used with Sapera LT 5.0 and the CamExpert utility.

CCA File Details

Teledyne DALSA distributes camera files using the ".CCA" extension, (CORECO CAMERA files), which contain all parameters describing the camera video signal characteristics and operation modes (what the camera outputs). The Sapera parameter groups within the file are:

- Video format and pixel definition.
- Video resolution (pixel rate, pixels per line, lines per frame).
- Synchronization source and timing.
- Channels/Taps configuration.
- Supported camera modes and related parameters.
- External signals assignment.

CVI File Details

Legacy files using the ".CVI" extension, (CORECO VIDEO files), contain all operating parameters related to the frame grabber board - what the frame grabber can actually do with camera controls or incoming video. The Sapera parameter groups within the file are:

- Activate and set any supported camera control mode or control variable.
- Define the integration mode and duration.
- Define the strobe output control.
- Allocate the frame grabber transfer ROI, the host video buffer size and buffer type (RGB888, RGB101010, MONO8, MONO16).

- Configuration of line/frame trigger parameters such as source (internal via the frame grabber /external via some outside event), electrical format (TTL, LVDS, OPTO-isolated), and signal active edge or level characterization.

Saving a Camera File

Use CamExpert to save a camera file (*.ccf) usable with any Sapera demo program or user application. An example would be a camera file, which sets up parameters for a free running camera (i.e. internal trigger) with exposure settings for a good image with common lighting conditions.

When CamExpert is setup as required, click on **File•Save As** to save the new .ccf file. The dialog that opens allows adding details such as camera information, mode of operation, and a file name for the .ccf file. The following image is a sample for a Teledyne DALSA Falcon camera. Note the default folder where User camera files are saved.

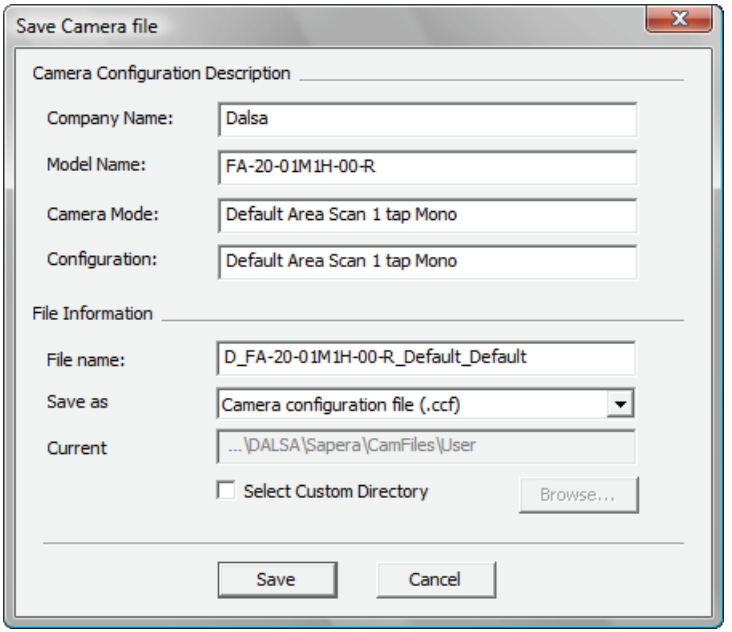


Figure 15: Saving a New Camera File (.ccf)

Camera Interfacing Check List

Before undertaking the task of interfacing a camera from scratch with CamExpert:

- Confirm that Teledyne DALSA has not already published an application note with camera files [http://www.teledynedalsa.com/mv/support].
- Confirm that the correct version or board revision of X64-CL is used. Confirm that the required firmware is loaded into the X64-CL.

- Confirm that Sapera does not already have a .cca file for your camera installed on your hard disk. If there is a .cca file supplied with Sapera, then use CamExpert to automatically generate the .ccf file with default parameter values matching the frame grabber capabilities.
- Check if the Sapera installation has a similar type of camera file. A similar .cca file can be loaded into CamExpert where it is modified to match timing and operating parameters for your camera, and lastly save them as Camera Configuration file (.ccf), or as a new .cca & .cvi camera file pair for applications built with Sapera 4.2 or earlier.
- Finally, if your camera type has never been interfaced, run CamExpert after installing Sapera and the acquisition board driver, select the board acquisition server, and enter the camera parameters.

Linescan Example: Interfacing the Teledyne DALSA Piranha2 Linescan Camera

These examples use a X64-CL Full board connected to the Piranha2 Camera Link camera (P2-2x or P2-4x). The model P2-4x 06k, a 4 tap 40 MHz 6k pixel digital linescan camera was used for the CamExpert screen shots. Download the user's manual directly from http://www.teledynedalsa.com/mv/support.

CamExpert Interfacing Outline

- Sapera and the X64-CL device driver are installed as described in this manual.
- Check for an existing CCA file, distributed with Sapera, that will provide camera timing parameters.
- Check the Teledyne DALSA web site for a published application note with camera files [http://www.teledynedalsa.com/mv/support].
- Connect two (for this example) Camera Link interface cables between the X64-CL and camera. Connect power to the camera.
- Run Windows HyperTerminal and establish communication with the camera. Verify the camera's settings (the Teledyne DALSA command to return all camera parameters is get_camera_parameters).
- Set the camera to its free run mode or its self generated test pattern. This will simplify testing the camera file timing parameters.
- Run CamExpert, load or set camera timing parameters and capabilities, and then test a live grab. Save the camera file for the default free run mode.
- Using HyperTerminal, set the camera to external sync and external line integration mode.
- Configure CamExpert for camera external sync with external line integration control. Test variations in line rate and integration period. Save a new camera file when satisfied.

Step 1: Piranha2 in Free Run Exposure Mode

This section illustrates the CamExpert dialog screens for interfacing the Teledyne DALSA Piranha2 in free run mode. Teledyne DALSA defines this exposure mode as where the camera uses its internal SYNC and PRIN, at a maximum line rate and exposure time. This is the Piranha2 factory default mode and serves well as a CamExpert interfacing example.

To verify or set the Piranha2 in free run mode:

- With HyperTerminal type the Teledyne DALSA command to return the camera parameters get_camera_parameters.
- Check the value for Exposure Mode, which is 2 for internal SYNC and PRIN.
- If the value is different, enter the command set_exposure_mode 2.

File Selection & Grab Test

- From the Windows start menu run the Sapera CamExpert program. [**Programs|Teledyne DALSA|Sapera LT|CamExpert**]
- CamExpert opens with default settings for the X64-CL (assuming the X64-CL is the only Sapera frame grabber installed). In the Device Selector area, click on the Camera Link Full Mono #1 acquisition server.
- The Configuration dro[p menu shows camera files distributed with Sapera that are supported by the X64-CL. The User's Configuration File section is the default location for *.ccf files saved by CamExpert and also the default location that Sapera demo programs use to read camera configuration files.
- Select the Teledyne DALSA P2-4x 06k Piranha2 8-bit camera. CamExpert automatically loads the basic timing parameters from the distribution cca file.
- With the Piranha2 configured for internal sync and PRIN (free run mode), click the Grab button to have live acquisition in the display window. The camera may need to be pointed to a bright wall or light source.
- Refer to the CamExpert video **status bar** to see if any required timing signals are missing. Also check the X64-CL status LEDs (see "Status LEDs Functional Description" on page 86) to aid in troubleshooting camera problems.
- This non-triggered exposure mode can be used to confirm a linescan inspection setup if the speed of the object or web is varied to match the fixed camera exposure.
- Use the **File-Save** dialog to save this camera configuration file (*.ccf) with user entered information. CamExpert provides information for each field based on the file originally loaded. Modify the fields such as camera mode and board configuration, to describe the parameter setup. Modify the file name and click Save. The ccf file is located in the default Sapera user folder.

Overview of Basic Timing Parameters

CamExpert only shows parameters applicable to the acquisition board and camera type. When configuring parameters for a new camera start by selecting or entering the basic horizontal timing parameters and pixel clock frequency as defined by the camera manufacturer.

The following screen image shows the Basic Timing Parameters required for the X64-CL with a linescan camera (parameters are for the Piranha2 - P2-4x 06k 8-bit camera).

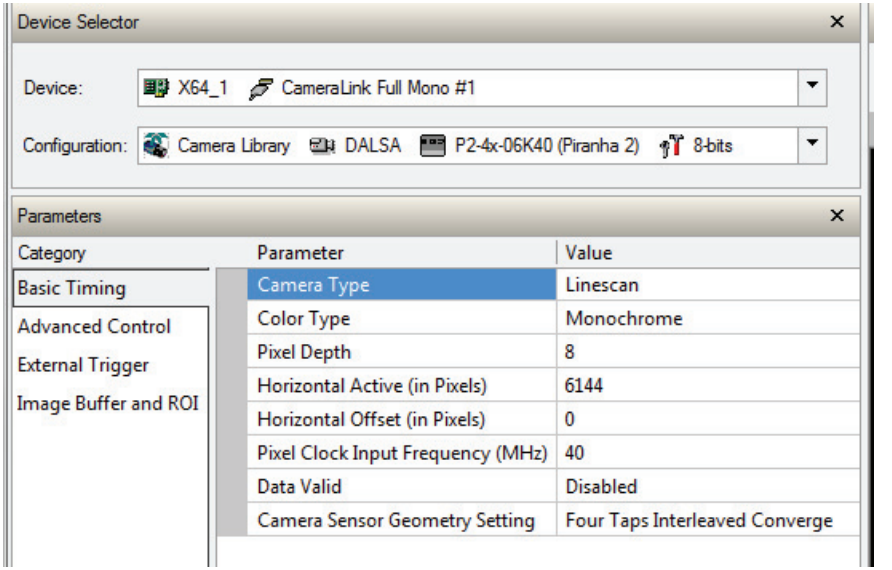


Figure 16: CamExpert - Camera Basic Timing

- Pixel Depth depends on the camera digital data. The Teledyne DALSA Piranha2 linescan camera digitizes internally to 10 bits and outputs either all 10 bits or the most significant 8 bits. When selecting 8 bits, the Sapera frame buffer required is 8 bit mono. When selecting 10 bit, the Sapera frame buffer required is 16 bit mono.
- Data Valid: Some Camera Link cameras use a data valid signal in addition to line valid and/or frame valid control signals. The Piranha2 does not, therefore this selection is set to disable.
- For the **Camera Sensor Geometry** parameter select one from the many standards supported by the acquisition board, or the user defines a custom geometry. The following screen image shows the sensor geometry selection window for the X64-CL.

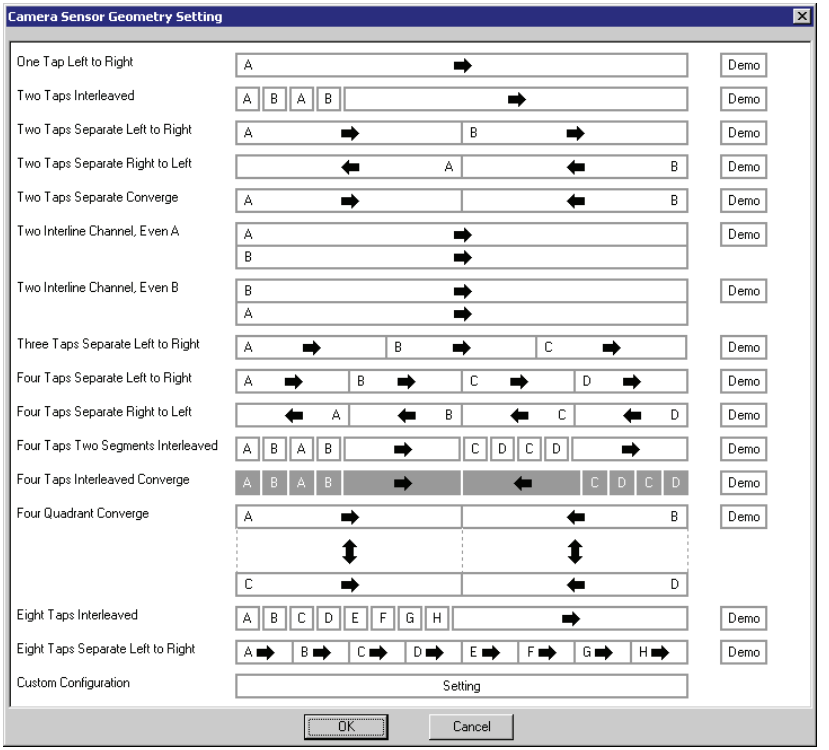

Camera Sensor Geometry Setting
One Tap Left to Right
Two Taps Interleaved
Two Taps Separate Left to Right
Two Taps Separate Right to Left
Two Taps Separate Converge
Two Interline Channel, Even A
Two Interline Channel, Even B
Three Taps Separate Left to Right
Four Taps Separate Left to Right
Four Taps Separate Right to Left
Four Taps Two Segments Interleaved
Four Taps Interleaved Converge
Four Quadrant Converge
Eight Taps Interleaved
Eight Taps Separate Left to Right
Custom Configuration
Setting
Demo
OK
Cancel

Figure 17: CamExpert – Sensor Geometry

The Piranha2 model P2-4x 06k, 4 tap - 6k pixel camera uses a tap structure as shown in the next figure. Tap 1 and 2 are simultaneously readout from a left to right direction while taps 3 and 4 are readout from a right to left direction. The advantage of multiple tap cameras is that the pixel clock rate is kept reasonably low while the data output to the frame grabber is increased. A camera with more than one tap has a higher average bandwidth by simultaneously outputting separate portions of a single sensor exposure.

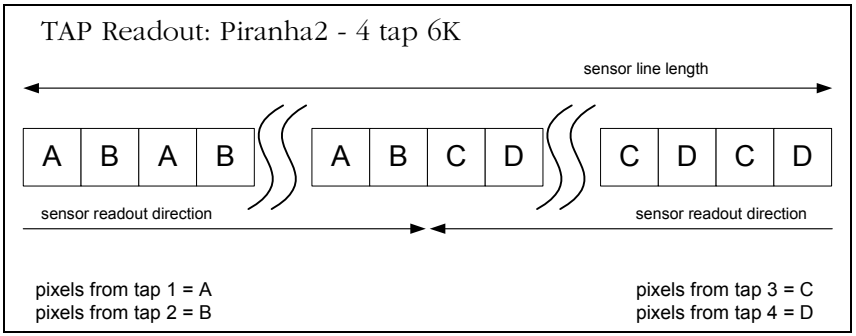

TAP Readout: Piranha2 - 4 tap 6K
sensor line length
A B A B
A B C D
C D C D
sensor readout direction
sensor readout direction
pixels from tap 1 = A
pixels from tap 2 = B
pixels from tap 3 = C
pixels from tap 4 = D

Figure 18: Piranha 2 – 4 tap description

Step 2: Piranha2 in External Exposure Mode

Using the Piranha2 camera in external exposure (line integration) mode requires a few changes to both the camera's operating mode and the Piranha2 camera file used for free run exposure mode. The modified camera file should be given an appropriate description and saved with a unique file name.

Before using CamExpert to generate and test the modified camera file, set the Piranha2 camera to the desired external exposure mode as follows:

- With HyperTerminal type the Teledyne DALSA command set_exposure_mode 5 to set the camera for external SYNC and PRIN control from the frame grabber. Acquisition and exposure will now be controlled by the X64-CL.
- The Piranha2 camera will respond with warnings that external SYNC and PRIN are not detected. This is normal since CamExpert has not yet been configured for line integration mode.

To simplify this example the X64-CL will generate the line sync trigger. The X64-CL is programmed for the line trigger frequency and line integration method matching the Piranha2 specifications.

CCF File Selection

To configure this new CCF file, start by loading the CCF for free run mode from the previous section.

- From the Windows start menu run the Sapera CamExpert program.
  [**All Programs**|**Teledyne DALSA**|**Sapera LT**|**CamExpert**]
- In the **Board** window, click on the Camera Link Full Mono #1 acquisition server.
- From the **Camera** file selection window, select the ccf camera file previously saved for the Piranha2 in free run exposure mode, as configured in the previous section "Step 1: Piranha2 in Free Run Exposure Mode" on page 41.

Advanced Control Parameters

The CamExpert advanced control parameters tab, contains the configuration items needed for external trigger and exposure control. Descriptions of each parameter, as setup for this example with the Piranha2, follow the screen image.

- **Line Sync Source**: Set to internal line trigger. The X64-CL will generate the camera line trigger without using an external event or trigger.
- **Internal Line Trigger Frequency**: Set the line rate generated by the X64-CL. The line sync rate is set to 4 kHz which is a period of 250µs. This period must be longer then the combined PRIN control pulse time (active low for pixel reset) and the integration time (time between rise of PRIN to EXSYNC). These control signals are described below.
- **Camera Line Trigger Frequency Min & Max**: Set to the camera specification limits so that CamExpert can trap invalid entries.
- **Enable Control Method**: Line Integration is selected as the required control mode.
- **Line Integration Method Setting and Camera Link Control Signals CC1-CC4**: The Camera Link CC1 and CC2 controls are assigned to Sapera controls which are not very descriptive by themselves. These selections are dependent on the camera's control specifications. The required camera controls

are matched to the corresponding Sapera exposure method. The logical sequence to set these parameters is as follows.

- From the Piranha2 user manual, the required Camera Link camera control configuration is **CC1 = EXSYNC** and **CC2 = PRIN**.
- From the Piranha2 user manual, note the timing requirements for external EXSYNC and PRIN. PRIN specifies a minimum logic low time for pixel reset which must be respected. When PRIN is logic high, the camera is integrating. Following an integration period the falling edge of EXSYNC triggers the line readout.
- Review the Sapera integration methods for a two pulse control which matches the camera control specifications. The Sapera Line Integration Method #1 matches the control requirement.
- From this information, it is seen that PRIN (CC2) corresponds to the Sapera control Pulse #0 and EXSYNC (CC1) corresponds to Sapera control Pulse # 1. Set the CamExpert camera controls to match these requirements. Note that Camera Link controls CC3 and CC4 are not used by the Piranha2.
- Click on the line integration method setting field. A configuration window allows selecting a method number corresponding to the Sapera integration methods.
- Select method 1. Configure the control pulses for polarity and signal width as shown in the following screen image. Descriptions for these parameters follow.

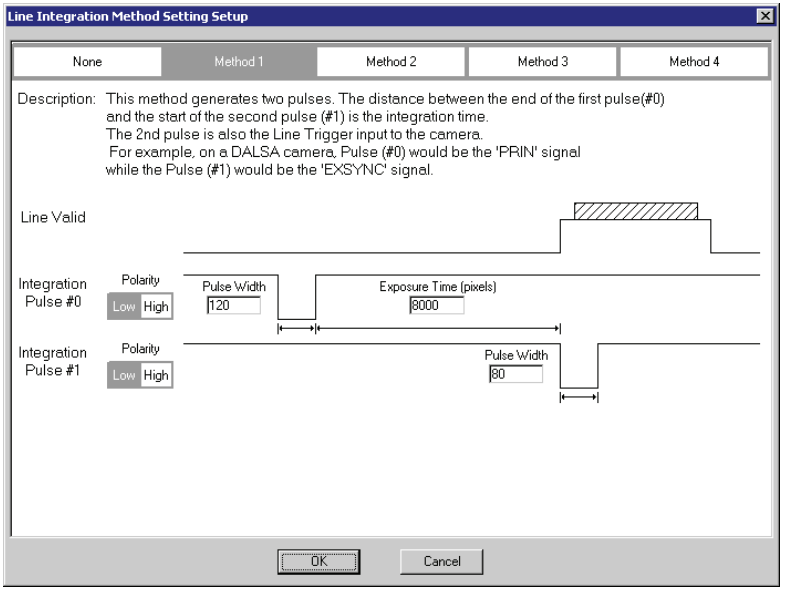

Line Integration Method Setting Setup
None
Method 1
Method 2
Method 3
Method 4
Description: This method generates two pulses. The distance between the end of the first pulse(#0) and the start of the second pulse (#1) is the integration time. The 2nd pulse is also the Line Trigger input to the camera. For example, on a DALSA camera, Pulse (#0) would be the 'PRIN' signal while the Pulse (#1) would be the 'EXSYNC' signal.
Line Valid
Integration Pulse #0
Polarity
Low
High
Pulse Width
120
Exposure Time (pixels)
8000
Integration Pulse #1
Polarity
Low
High
Pulse Width
80
OK
Cancel

Figure 19: Line Integration Method setup

- For Pulse #0 (assigned to PRIN), select polarity and width. The Piranha2 specifies that PRIN must be a minimum 2µs active low pulse. At a 40Mhz pixel clock, 40 pixels equal 1µs, thus a setting of 120 pixels generates a 3µs PRIN. Note that the X64-CL on-board pulse generator works in increments of 1µs.

- Knowing that 40 pixels is 1µs, this example sets an integration time of 8000 pixels which is 200µs. This value must be less then the line trigger rate which was set to 4 KHz (250µs).
- For Pulse #1 (assigned to EXSYNC), select polarity and width. In this example EXSYNC is active low for 2 µs (80 pixels).

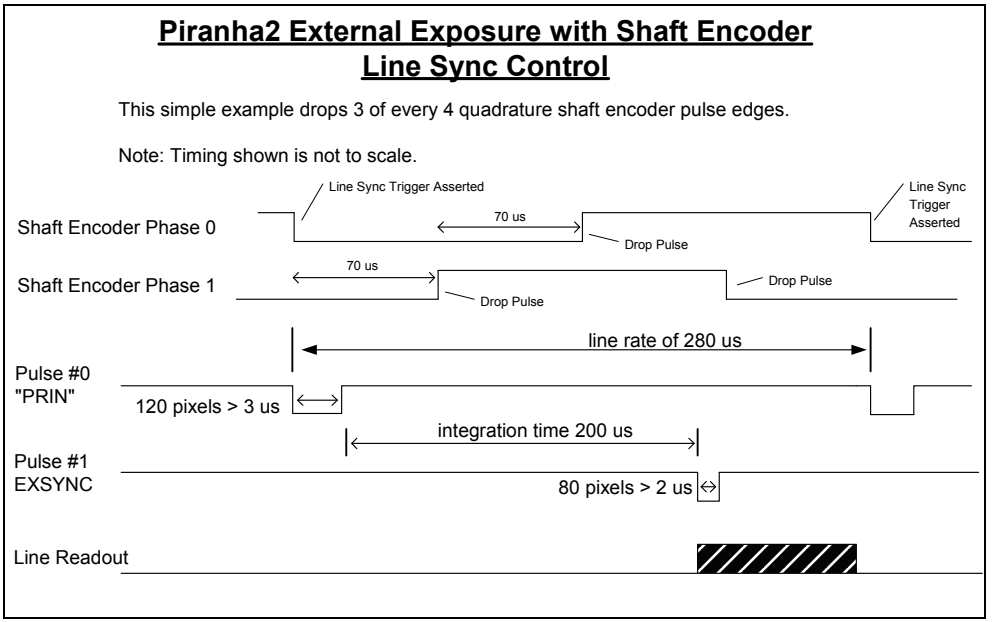
Step 3: Piranha2 with Shaft Encoder Line Sync

Continuing from the previous setup ("Step 2: Piranha2 in External Exposure Mode" on page 44), this section details using the X64-CL shaft encode inputs as the exposure trigger for the imaging system. In addition the virtual frame reset feature is used to have an n number of image lines grabbed into the Sapera frame buffer when triggered by some external event.

- See "Shaft Encoder Interface Timing" on page 58 for an overview of the quadrature shaft encoder supported by the X64-CL, and the connections used.
- See "Virtual Frame Trigger for Linescan Cameras" on page 59 for an overview of using an external frame reset signal to initiate the acquisition of n number of lines, and the connections used.

Shaft Encoder Line Sync Setup

- Assuming the same PRIN and EXSYNC timing from the previous example, on the Advanced Control Parameters tab, the **Line Sync Source** signal Shaft Encoder is now selected.
- This example uses a line integration time of 200µs therefore the line sync source must have a period greater then the integration time plus the PRIN duration.
- Assuming quadrature shaft encoder signals every 70µs, by dropping three out of every four pulses, a line sync occurs every 280µs. The following timing diagram (not to scale) illustrates the shaft encoder signals relative to PRIN and EXSYNC.



- Select the CamExpert **External Trigger Parameters** tab. The External Line Trigger Source is set to use both shaft encoder inputs. The edge drop factor is set to 3 to match the example timing diagram shown above.

Using shaft encoder signals only, the X64-CL grabs linescan data continuously into a memory frame buffer. The number of data lines stored (i.e. the vertical size of this frame buffer – see the Image Height parameter on the Image Buffer tab) is arbitrarily set by the user. With continuous shaft encode signals, after the frame buffer is filled with captured data lines, new data lines will then overwrite previous data.

Shaft Encoder with Fixed Frame Buffer Setup

To synchronize the capture of linescan data, an external trigger signal input to the X64-CL is used to start acquisition into the frame buffer. In this example, when the external trigger is asserted, the X64-CL then triggers linescan data captures based on the shaft encoder inputs. When the frame buffer is filled, linescan capture is suspended until the next external trigger. Example screen images of the External Trigger Parameters and the Image Buffer Parameters tabs follow.

- **External Trigger**: Set to Enable
- **External Trigger Detection**: Select which pulse edge or from a level active trigger (100ns minimum pulse width). Available choices are dependent on the acquisition board used.
- **External Trigger Level**: Choose the signal type connect as an external trigger.
- **Image Width**: By default the buffer is the same as the acquisition width.
- **Image Height**: Set to the maximum number of acquired lines needed. Refer to the hardware specifications for the maximum limit.

- **Acquisition Frame Length method**: In this example, set to fixed length when there is a single trigger signal to start the acquisition of one complete frame buffer.

Assuming that the shaft encoder and external trigger signals are connected (refer to ": External Signals Connector Bracket Assembly" on page 93), click on Grab to test the linescan camera setup. Note that CamExpert includes a **"soft" trigger** button in the Display window, which simulates an external trigger event.

Shaft Encoder with Variable Frame Buffer Setup

To synchronize the capture of a variable amount of linescan data, external trigger signals input to the X64-CL are used to start and stop line acquisition into the frame buffer. The grab control can either be a level type (where the acquisition occurs while the external control is active high), or the grab control can be a two pulse trigger (where trigger pulse 1 starts the line acquisition and trigger pulse 2 stops acquisition.

The actual linescan data capture is still triggered by the shaft encoder inputs. If the frame buffer is filled before the stop acquisition trigger control occurs, the linescan capture is suspended until the next start acquisition trigger. Example screen images of the External Trigger Parameters and the Image Buffer Parameters tabs follow.

- **External Trigger**: Set to Enable
- **External Trigger Detection**: In this example trigger input 1 starts the linescan acquisition and trigger input 2 ends the acquisition. The number of lines is variable.
- **External Trigger Level**: Choose the signal type connect as an external trigger.
- **Image Width**: By default the buffer is the same as the acquisition width.
- **Image Height**: Set to the maximum number of acquired lines needed. Refer to the hardware specifications for the maximum limit.
- **Acquisition Frame Length method**: In this example, set to variable length when there is both a start trigger pulse and an end trigger pulse to control the acquisition into the frame buffer. A level trigger signal (active high period or active low period) is also used for a grab controller with a variable length frame buffer.

Assuming that the shaft encoder and external trigger signals are connected (refer to ": External Signals Connector Bracket Assembly" on page 93), click on Grab to test the linescan camera setup. Note that CamExpert includes a **"soft" trigger** button in the Display window, which simulates an external trigger event.

Sapera Demo Applications

Grab Demo Overview

Sapera comes with a number of demo applications. A simple acquisition demo is described below.

Program	**Start • All Programs • TELEDYNE DALSA • Sapera LT • Demos•Frame Grabbers • GrabDemo**
Program file	**…\TELEDYNE DALSA\Sapera\Demos\Classes\vc\GrabDemo\Release\GrabDemo.exe**
Description	This program demonstrates the basic acquisition functions included in the Sapera library. The program allows you to acquire images, either in continuous or in one-shot mode, while adjusting the acquisition parameters. The program code may be extracted for use within your own application.
Remarks	This demo is built using Visual C++ 6.0 using the MFC library. It is based on Sapera C++ classes. See the Sapera User's and Reference manuals for more information.

Using the Grab Demo

Server Selection

The demo program first displays the acquisition configuration menu. The first drop menu displayed permits selecting from any installed Sapera acquisition servers (installed Teledyne DALSA acquisition hardware using Sapera drivers). The second drop menu permits selecting from the available input devices present on the selected server.

CCF File Selection

The acquisition configuration menu is also used to select the required camera configuration file for the connected camera. Sapera camera files contain timing parameters and video conditioning parameters. The default folder for camera configuration files is also used by the CamExpert utility to save user generated or modified camera files.

Use the Sapera CamExpert utility program to generate the camera configuration file based on timing and control parameters entered. The CamExpert live acquisition window allows immediate verification of those parameters. CamExpert reads both Sapera *.cca and *.cvi for backward compatibility with the original Sapera camera files.

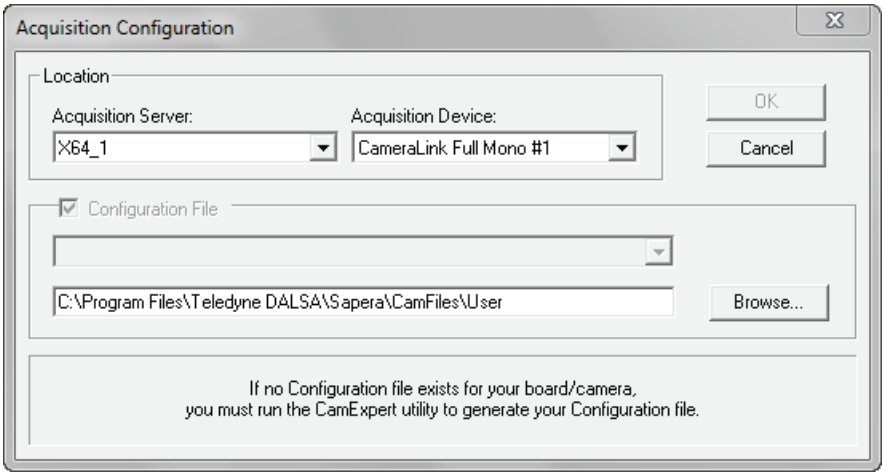


Figure 20: Camera Configuration File selection

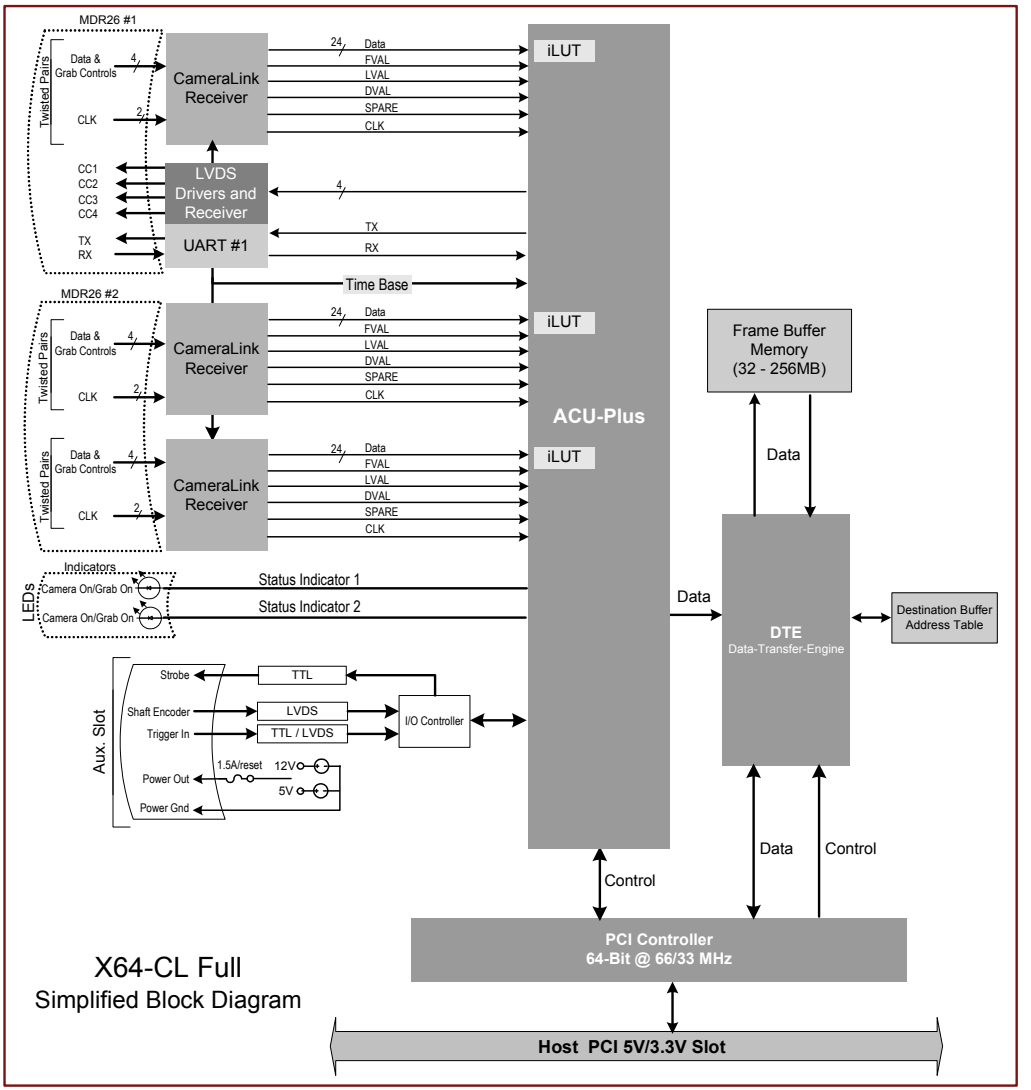
Grab Demo Main Window

The demo main window provides control buttons and a central area for displaying the grabbed image. Developers can use the demo source code as a foundation to quickly create and test the desired imaging application.

Refer to the Sapera LT User's Manual (OC-SAPM-USER), in section "Demos and Examples – Acquiring with Grab Demo", for more information on the Grab Demo.

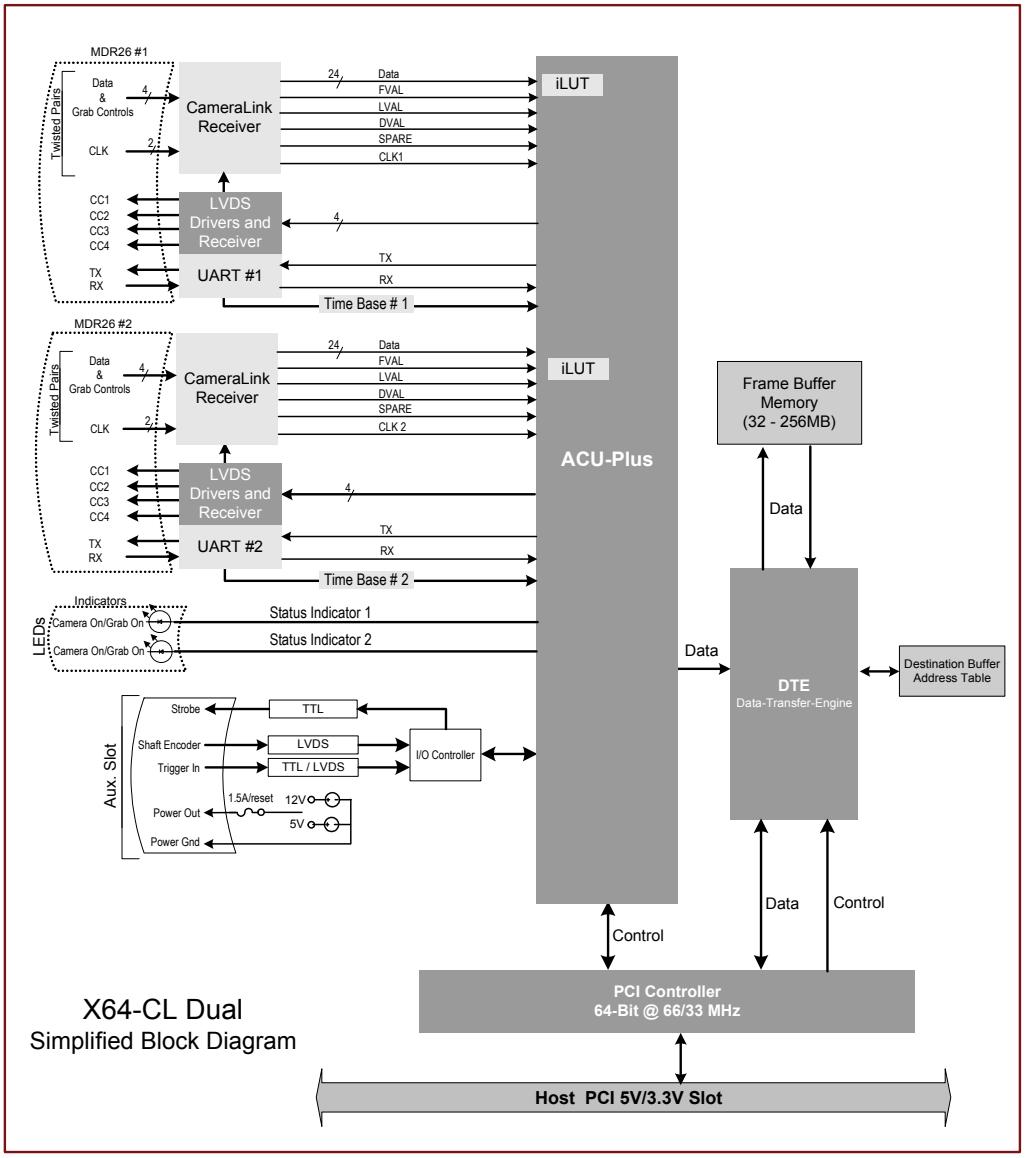
X64-CL Reference

X64-CL Full Block Diagram

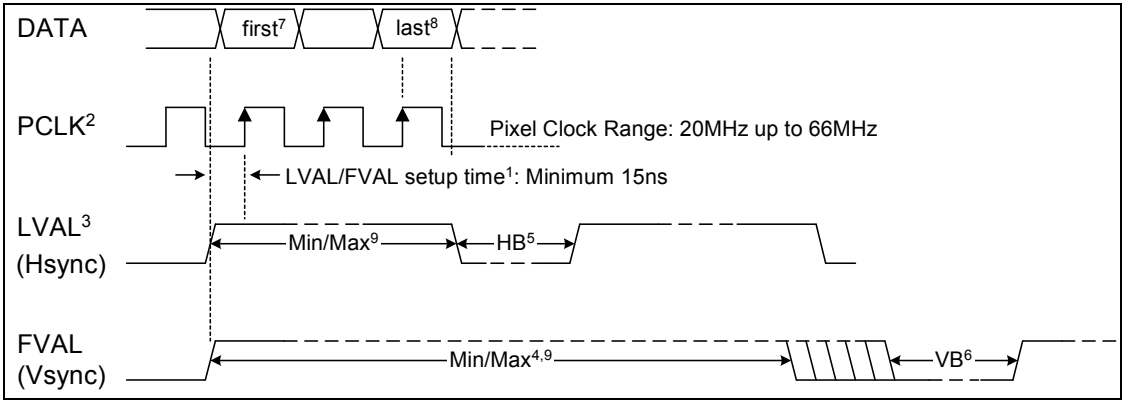


X64-CL Full
Simplified Block Diagram

X64-CL Dual Block Diagram



X64-CL Acquisition Timing



- [1] The setup times for LVAL and FVAL are the same. Both must be high and stable before the rising edge of the Pixel Clock.
- [2] Pixel Clock must always be present.
- [3] LVAL must be active high to acquire camera data.
- [4] Minimum of 1.
- [5] HB - Horizontal Blanking:

  Minimum: 4 clock cycles

  Maximum: no limits

- [6] VB - Vertical Blanking:

  Minimum: 1 line

  Maximum: no limits

- [7] First Active Pixel (unless otherwise specified in the CCA file – "Horizontal Back invalid = x" where 'x' defines the number of pixels to be skipped).
- [8] Last Active Pixel – defined in the CCA file under "Horizontal active = y" – where 'y' is the total number of active pixels per tap.
- [9] Maximum Valid Data:
  - 8-bits/pixel x 256K Pixels/line (LVAL)
  - 16-bits/pixel x 128K Pixels/line (LVAL)
  - 32-bits/pixel x 64K Pixels/line (LVAL)
  - 64-bits/pixel x 32K Pixels/line (LVAL)
  - 16,000,000 lines (FVAL)

External Frame Trigger Source Selection for Area Scan Application

Area scan imaging applications often require some form of external event frame trigger to synchronize camera exposures. This synchronization signal is either an external trigger source (minimum trigger pulse = 100ns) or an internal board sync.

The following table describes the frame trigger source types supported by either the X64-CL Full or X64-CL Dual boards. Refer to the Sapera Acquisition Parameters Reference Manual (OC-SAPM-APR00) for descriptions of the Sapera parameters.

Parameter Values Specific to the X64-CL

CORACQ_PRM_EXT_FRAME_TRIGGER_SOURCE and
CORACQ_PRM_EXT_TRIGGER_SOURCE – Parameter Values Specific to the X64-CL

PRM Value	Active External Trigger Input
0	Default
1	Use External Trigger 1
2	Use External Trigger 2
3	Use Board Sync
4	Output to Board Sync

Detailed Description

PRM Value	X64-CL model & camera input used	External Frame Trigger Signal used or External Trigger Signal used
0	Dual - Camera #1	External Trigger 1
	Dual - Camera #2	External Trigger 2
	Full - Camera #1	External Trigger 1
1	Dual - Camera #1	External Trigger 1
	Dual - Camera #2	External Trigger 1
	Full - Camera #1	External Trigger 1
2	Dual - Camera #1	External Trigger 2
	Dual - Camera #2	External Trigger 2
	Full - Camera #1	External Trigger 2
3	Dual - Camera #1	Board Sync
	Dual - Camera #2	Board Sync
	Full - Camera #1	Board Sync
4	Dual - Camera #1 *	External Trigger 1
	Dual - Camera #2 *	External Trigger 2
	Full - Camera #1	External Trigger 1

Note (*): Dual Camera #1 and Dual Camera #2 cannot have the parameter set to 4 at the same time.

See "J4 (CL or EM rev. A0, A1): External Signals Connector " on page 89 for external trigger input connector details.

CVI/CCF File Parameters Used

- External Trigger Source = prm value
- External Trigger Enable = CORACQ_VAL_EXT_TRIGGER_OFF (0x00000001) or CORACQ_VAL_EXT_TRIGGER_ON (0x00000008)

- External Frame Trigger Source = prm value
- External Frame Trigger Enable = TRUE (0x00000001) or FALSE (0x00000000)

Line Trigger Source Selection for Linescan Applications

Linescan imaging applications require some form of external event trigger to synchronize linescan camera exposures to the moving object. This synchronization signal is either an external trigger source (minimum trigger pulse = 100ns) (one exposure per trigger event) or a shaft encoder source composed of a single or dual phase (quadrature) signal. The X64-CL shaft encoder inputs provide additional functionality with pulse drop or pulse multiply support.

The following table describes the line trigger source types supported by either the X64-CL Full or X64-CL Dual boards. Refer to the Sapera Acquisition Parameters Reference Manual (OC-SAPM-APR00) for descriptions of the Sapera parameters.

Parameter Values Specific to the X64-CL

CORACQ_PRM_EXT_LINE_TRIGGER_SOURCE – Parameter Values Specific to the X64-CL

PRM Value	Active Shaft Encoder Input
0	Default
1	Use phase A
2	Use phase B
3	Use phase A & B

CORACQ_PRM_EXT_LINE_TRIGGER_SOURCE full description relative to trigger type and X64-CL model used:

PRM Value	X64-CL model & camera input used	External Line Trigger Signal used	External Shaft Encoder Signal used
		if CORACQ_PRM_EXT_LINE_ TRIGGER_ENABLE = *true*	*if* CORACQ_PRM_SHAFT_ ENCODER_ENABLE =*true*
0	Dual - Camera #1	Shaft Encoder Phase A	Shaft Encoder Phase A
	Dual - Camera #2	Shaft Encoder Phase B	Shaft Encoder Phase B
	Full - Camera #1	Shaft Encoder Phase A	Shaft Encoder Phase A & B
1	Dual - Camera #1	Shaft Encoder Phase A	Shaft Encoder Phase A
	Dual - Camera #2	Shaft Encoder Phase A	Shaft Encoder Phase A
	Full - Camera #1	Shaft Encoder Phase A	Shaft Encoder Phase A
2	Dual - Camera #1	Shaft Encoder Phase B	Shaft Encoder Phase B
	Dual - Camera #2	Shaft Encoder Phase B	Shaft Encoder Phase B
	Full - Camera #1	Shaft Encoder Phase B	Shaft Encoder Phase B
3	Dual - Camera #1	n/a	Shaft Encoder Phase A & B
	Dual - Camera #2	n/a	Shaft Encoder Phase A & B
	Full - Camera #1	n/a	n/a – use prm value = 0

See "J4 (CL or EM rev. A0, A1): External Signals Connector " on page 89 for shaft encoder input connector details.

CVI/CCF File Parameters Used

- External Line Trigger Source = prm value
- External Line Trigger Enable = true/false
- Shaft Encoder Enable = true/false

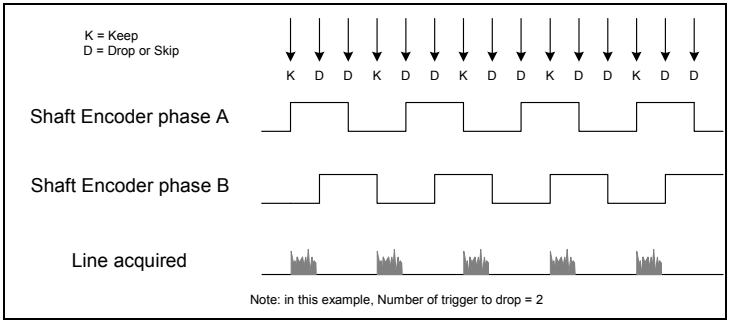
Shaft Encoder Interface Timing

Connector J4, Dual Balanced Shaft Encoder Inputs:

- Input 1: Pin 15 (Phase A +) & Pin 16 (Phase A -)
  (see "J4 (CL or EM rev. A0, A1): External Signals Connector " on page 89 for complete connector signal details)
- Input 2: Pin 17 (Phase B +) & Pin 18 (Phase B -)
- For X64-CL or X64-EM rev. A0, A1: use external signals cable assembly OC-64CC-0TIO1
  (see "X64-CL or X64-EM rev. A0, A1: External Signals Connector Bracket Assembly" on page 93 for pinout)
- For X64-EM rev. A2 or later: use external signals cable assembly OC-64CC-GIO25EM
  (see "X64-EM rev. A2 & later: External Signals Connector Bracket Assembly" on page 97 for pinout)

Web inspection systems with variable web speeds typically provide one or two synchronization signals from a web mounted encoder to coordinate trigger signals. These trigger signals are used by the acquisition linescan camera. The X64-CL supports single or dual shaft encoder signals. Dual encoder signals are typically 90 degrees out of phase relative to each other and provide greater web motion resolution. When using only one shaft encoder input phase, say phase A, then the phase B inputs must be terminated by connecting the + input to a voltage a minimum of 100 mV positive relative to the – input.

When enabled, the camera is triggered and acquires one scan line for each shaft encoder pulse edge. To optimize the web application, a second Sapera parameter defines the number of triggers to skip between valid acquisition triggers. The figure below depicts a system where a valid camera trigger is any pulse edge from either shaft encoder signal. After a trigger the two following triggers are ignored (as defined by a Sapera parameter).


K = Keep
D = Drop or Skip
K D D K D D K D D K D D K D D
Shaft Encoder phase A
Shaft Encoder phase B
Line acquired
Note: in this example, Number of trigger to drop = 2

Note that camera file parameters are best modified by using the Sapera CamExpert program.

CVI/CCF File Parameters Used

Shaft Encoder Enable = X, where:

- If X = 1, Shaft Encoder is enabled
- If X = 0, Shaft Encoder is disabled

Shaft Encoder Pulse Drop = X, where:

- X = number of trigger pulses ignored between valid triggers

For information on camera configuration files see the Sapera Acquisition Parameters Reference Manual (OC-SAPM-APR00).

Virtual Frame Trigger for Linescan Cameras

When using linescan cameras a frame buffer is allocated in host system memory to store captured video lines. To control when a video line is stored as the first line in this "virtual" frame buffer, an external frame trigger signal is used. The number of lines sequentially grabbed and stored in the virtual frame buffer is controlled by the Sapera vertical cropping parameter.

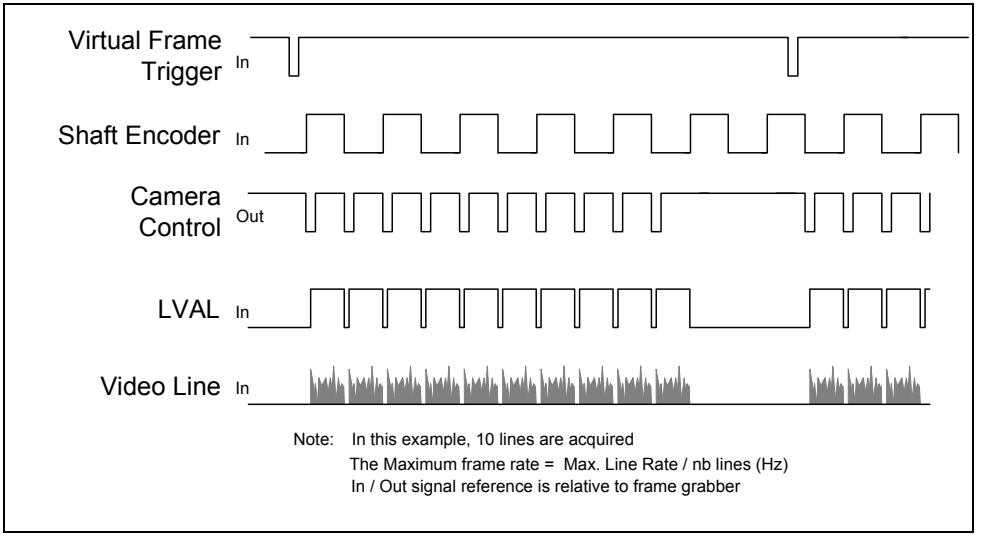
Virtual Frame Trigger Timing Diagram

The following timing diagram shows an example of grabbing 10 video lines from a line scan camera and the use of virtual frame trigger to define when a video line is stored at the beginning of the virtual frame buffer. The virtual frame trigger signal (generated by some external event) is input on the X64-CL trigger input.

- **Virtual frame trigger** can be TTL or LVDS and be rising or falling edge active.
- **Virtual frame trigger** control is configured for rising edge trigger in this example.
- **Virtual frame trigger** connects to the X64-CL via the Trigger In 1 balanced inputs on connector J4 pin 11 (+) and 12 (-).
- Camera control signals are active at all times. These continually trigger the camera acquisition in order to avoid corrupted video lines at the beginning of a virtual frame.
- The camera control signals can be based on the X64-CL shaft encoder inputs, or line triggers generated internally by the X64-CL.
- The number of lines captured is specified by the Sapera vertical cropping parameter.

Synchronization Signals for a Virtual Frame of 10 Lines.

The following timing diagram shows the relationship between external Frame_Reset input, external Shaft Encoder input (one phase used with the second terminated), and EXSYNC out to the camera.



CVI File (VIC) Parameters Used

The VIC parameters listed below provide the control functionality for virtual frame reset. Applications either load pre-configured .cvi files or change VIC parameters directly during runtime.

Note that camera file parameters are best modified by using the Sapera CamExpert program.

External Frame Trigger Enable = X, where: (with Virtual Frame Trigger enabled)

- If X = 1, External Frame Trigger is enabled
- If X = 0, External Frame Trigger is disabled

External Frame Trigger Detection = Y, where: (with Virtual Frame Trigger edge select)

- If Y= 4, External Frame Trigger is active on rising edge
- If Y= 8, External Frame Trigger is active on falling edge

External Frame Trigger Level = Z, where: \\ Frame_Reset signal type

- If Z= 2, External Frame Trigger is a LVDS signal

For information on camera files see the Sapera Acquisition Parameters Reference Manual (OC-SAPM-APR00).

Sapera Acquisition Methods

Sapera acquisition methods define the control and timing of the camera and frame grabber board. Various methods are available, grouped as:

- Camera Reset Methods (method 1 supported)
- Camera Trigger Methods (method 1 and 2 supported)
- Line Integration Methods (method 1 through 4 supported)
- Line Trigger Methods (method 1 supported)
- Time Integration Methods (method 1 through 8 supported)
- Strobe Methods (method 1 through 4 supported)

Refer to the Sapera LT Acquisition Parameters Reference manual (OC-SAPM-APR00) for detailed information concerning camera and acquisition control methods.

Trigger-to-Image Reliability

Trigger-to-image reliability incorporates all stages of image acquisition inside an integrated controller to increase reliability and simplify error recovery. The trigger-to-image reliability model brings together all the requirements for image acquisition to a central management unit. These include I/O signals to control camera timing, onboard frame buffer memory to compensate for PCI bus latency, and comprehensive error notification. If the X64-CL detects a problem, the application can be informed and take appropriate action to return to normal operation.

The X64-CL is designed with a robust ACU (Acquisition and Control Unit). The ACU monitors in real-time, the acquisition state of the input plus the DTE (Data Transfer Engine) which transfers image data from on-board memory into PC memory. In general, these management processes are transparent to end-user applications. With the X64-CL, applications ensure trigger-to-image reliability by monitoring events and controlling transfer methods as described below:

Trigger Signal Validity

External trigger signal noise or glitches are easily ignored by the ACU with its programmable debounce control. A parameter is programmed for the minimum pulse duration considered as a valid external trigger pulse.

Supported Events and Transfer Methods

Listed below are the supported acquisition and transfer events. Event monitoring is a major component to the Trigger-to-Image Reliability framework.

Acquisition Events

Acquisition events are related to the acquisition module. They provide feedback on the image capture phase.

- **External Trigger** (Used/Ignored)
  Generated when the external trigger pin is asserted, which indicates the start of the acquisition process. There are two types of external trigger events: 'Used' or 'Ignored'. Following an external trigger, if the event generates a captured image, an External Trigger Used event will be generated (CORACQ_VAL_EVENT_TYPE_EXTERNAL_TRIGGER).
  If there is no captured image, an External Trigger Ignored event will be generated (CORACQ_VAL_EVENT_TYPE_EXTERNAL_TRIGGER_IGNORED). An external trigger event is ignored if the event rate is higher than the possible frame rate of the camera.
- **Data Overflow**
  The Data Overflow event indicates that there is not enough bandwidth for the acquired data to be transferred without loss. This is usually caused by limitations of the acquisition module and should never occur.
  The Sapera event value is CORACQ_VAL_EVENT_TYPE_DATA_OVERFLOW.
- **Pixel Clock** (Present/Absent)
  Event generated on the transition from detecting or not detecting a pixel clock signal. The Sapera event values are CORACQ_VAL_EVENT_TYPE_NO_PIXEL_CLK and CORACQ_VAL_EVENT_TYPE_PIXEL_CLK.
- **Frame Lost**
  The Frame Lost event indicates that an acquired image could not be transferred to on-board memory. An example of this case would be if there are no free on-board buffers available for the new image. This may be the case if the image transfer from onboard buffers to host PC memory cannot be sustained due to bus bandwidth issues.
  The Sapera event value is CORACQ_VAL_EVENT_TYPE_FRAME_LOST.

Transfer Events

Transfer events are the ones related to the transfer module. Transfer events provide feedback on image transfer from onboard memory frame buffers to PC memory frame buffers.

- **Start of Frame**
  The Start of Frame event is generated when the first image pixel is transferred from onboard memory into PC memory.
  The Sapera event value is CORXFER_VAL_EVENT_TYPE_START_OF_FRAME.
- **End of Frame**
  The End of Frame event is generated when the last image pixel is transferred from onboard memory

into PC memory.
The Sapera event value is CORXFER_VAL_EVENT_TYPE_END_OF_FRAME.

- **End of Line**
  The End of Line event is generated after a video line is transferred to a PC buffer.
  The Sapera event value is CORXFER_VAL_EVENT_TYPE_END_OF_LINE.
- **End of N Lines**
  The End of N Lines event is generated after a set number of video lines are transferred to a PC buffer. The Sapera event value is CORXFER_VAL_EVENT_TYPE_END_OF_NLINES.
- **End of Transfer**
  The End of Transfer event is generated at the completion of the last image being transferred from onboard memory into PC memory. To complete a transfer, a stop must be issued to the transfer module (if transfers are already in progress). If a transfer of a fixed number of frames was requested, the transfer module will stop transfers automatically. The Sapera event value is CORXFER_VAL_EVENT_TYPE_END_OF_TRANSFER.

Trigger Signal Validity

The ACU ignores external trigger signal noise with its programmable debounce control. Program the debounce parameter for the minimum pulse duration considered as a valid external trigger pulse. Refer to "External Trigger TTL Input Electrical Specification" on page 102 for more information.

Supported Transfer Cycling Methods

The X64-CL supports the following transfer modes, which are either synchronous or asynchronous. These definitions are from the Sapera Basic Reference manual.

- CORXFER_VAL_CYCLE_MODE_SYNCHRONOUS_WITH_TRASH
  Before cycling to the next buffer in the list, the transfer device will check the next buffer's state. If its state is full, the transfer will be done in the trash buffer, which is defined as the last buffer in the list; otherwise, it will occur in the next buffer. After a transfer to the trash buffer is done, the transfer device will check again the state of the next buffer. If it is empty, it will transfer to this buffer otherwise it will transfer again to the trash buffer.
- CORXFER_VAL_CYCLE_MODE_SYNCHRONOUS_NEXT_EMPTY_WITH_TRASH
  Before cycling to the next buffer in the list, the transfer device will check the next buffer's state. If its state is full, the next buffer will be skipped, and the transfer will be done in the trash buffer, which is defined as the last buffer in the list; otherwise it will occur in the next buffer. After a transfer to the trash is done, the transfer device will check the next buffer in the list, if its state is empty, it will transfer to this buffer otherwise it will skip it, and transfer again to the trash buffer.
- CORXFER_VAL_CYCLE_MODE_ASYNCHRONOUS
  The transfer device cycles through all buffers in the list without concern about the buffer state.

Output LUT Availability

The following tables define the X64-CL input LUT availability and function modes. The first table is for the default firmware and the second for the optional 12-bit data with LUT firmware.

Firmware: Standard (default)

Number of Digital Bits	Number of Taps with LUT	Number of Taps Without LUT	Output Pixel Format	LUT Format	Notes
8	1, 2, 4	1, 2, 4, 8	Mono 8	8-in, 8-out	
8	1, 2, 4	1, 2, 4	Mono 16	8-in, 8-out	8 bits in 8 LSBs of 16-bit
10	1, 2, 4	1, 2, 4	Mono 8	10-in, 8-out	LUT is used to convert 10 --> 8
10	1, 2, 4	1, 2, 4	Mono 16	10-in, 10-out	10 bits in 10 LSBs of 16-bit
12	-	-	*Mono 8*	-	*Not supported with standard FPGA*
12	0	1, 2, 4	Mono 16	-	
14	-	-	*Mono 8*	-	*Not supported with standard FPGA*
14	0	1, 2, 4	Mono 16	-	
16	-	-	*Mono 8*	-	*Not supported with standard FPGA*
16	0	1, 2, 4	Mono 16	-	
8 x 3 (RGB)	1	2	RGB8888	8-in, 8-out	
8 x 3 (RGB)	-	-	*RGB101010*	-	*Not Supported*
10 x 3 (RGB)	1	1	RGB8888	10-in, 8-out	LUT is used to convert 10-->8
10 x 3 (RGB)	1	1	RGB101010	10-in, 10-out	

Notes:

- There is only 1 LUT per camera.
- Pixel data bit reduction is only available when the LUT is available.
- X = No LUT available.

Firmware: X64-CL-Full - 12-bit LUT

This table applies to a X64-CL Full board only, not a Dual Base board and not a X64-CL EM board.

Number of Digital Bits	Number of Taps with LUT	Output Pixel Format	LUT Format	Notes
8	-	MONO 8	-	Not Supported
8	-	MONO 16	-	Not Supported
10	-	MONO 8	-	Not Supported
10	-	MONO 16	-	Not Supported
12	2	MONO 8	12-in, 8-out	12 bits in – 8 MSB out to buffer
12	2	MONO 16	12-in, 12-out	12 bits in 12 LSBs of 16-bit
14	-	MONO 8	-	Not Supported
14	-	MONO 16	-	Not Supported
16	-	MONO 8	-	Not Supported
16	-	MONO 16	-	Not Supported
8 x 3 (RGB)	-	RGB8888	-	Not Supported
8 x 3 (RGB)	-	RGB101010	-	Not Supported
10 x 3 (RGB)	-	RGB888	-	Not Supported
10 x 3 (RGB)	-	RGB101010	-	Not Supported

Supporting Non-Standard Camera Link Cameras

High performance cameras that output 10 taps can not be interfaced with a standard Camera Link full specification frame grabber. The X64-CL (85 MHz version) provides support for a non-standard 10-tap format by using a specific firmware version, easily uploaded when required.

Firmware: X64-CL 10-Tap Format

- Requires X64-CL 85 MHz board
- Supports 10-tap Format cameras such as Basler A504K
- This Camera Link utilization is not compatible with the standard 8 tap full specification.
- No LUT available on the X64-CL.
- The following table describes the data Bit assignment.
- Tap 1 Bits are D0_x … Tap 10 Bits are D9_x

Connector 1 Channel Link No. X		Connector 2 Channel Link No. Y		Connector 2 Channel Link No. Z	
Bit Name	Input/Output Pin	Bit Name	Input/Output Pin	Bit Name	Input/Output Pin
D0_0	Tx0/Rx0	D3_2	Tx0/Rx0	D6_5	Tx0/Rx0
D0_1	Tx1/Rx1	D3_3	Tx1/Rx1	D6_6	Tx1/Rx1
D0_2	Tx2/Rx2	D3_4	Tx2/Rx2	D6_7	Tx2/Rx2
D0_3	Tx3/Rx3	D3_5	Tx3/Rx3	D7_0	Tx3/Rx3
D0_4	Tx4/Rx4	D3_6	Tx4/Rx4	D7_1	Tx4/Rx4
D0_5	Tx5/Rx5	D3_7	Tx5/Rx5	D7_2	Tx5/Rx5
D0_6	Tx6/Rx6	D4_0	Tx6/Rx6	D7_3	Tx6/Rx6
D0_7	Tx7/Rx7	D4_1	Tx7/Rx7	D7_4	Tx7/Rx7
D1_0	Tx8/Rx8	D4_2	Tx8/Rx8	D7_5	Tx8/Rx8
D1_1	Tx9/Rx9	D4_3	Tx9/Rx9	D7_6	Tx9/Rx9
D1_2	Tx10/Rx10	D4_4	Tx10/Rx10	D7_7	Tx10/Rx10
D1_3	Tx11/Rx11	D4_5	Tx11/Rx11	D8_0	Tx11/Rx11
D1_4	Tx12/Rx12	D4_6	Tx12/Rx12	D8_1	Tx12/Rx12
D1_5	Tx13/Rx13	D4_7	Tx13/Rx13	D8_2	Tx13/Rx13
D1_6	Tx14/Rx14	D5_0	Tx14/Rx14	D8_3	Tx14/Rx14
D1_7	Tx15/Rx15	D5_1	Tx15/Rx15	D8_4	Tx15/Rx15
D2_0	Tx16/Rx16	D5_2	Tx16/Rx16	D8_5	Tx16/Rx16
D2_1	Tx17/Rx17	D5_3	Tx17/Rx17	D8_6	Tx17/Rx17
D2_2	Tx18/Rx18	D5_4	Tx18/Rx18	D8_7	Tx18/Rx18
D2_3	Tx19/Rx19	D5_5	Tx19/Rx19	D9_0	Tx19/Rx19
D2_4	Tx20/Rx20	D5_6	Tx20/Rx20	D9_1	Tx20/Rx20
D2_5	Tx21/Rx21	D5_7	Tx21/Rx21	D9_2	Tx21/Rx21
D2_6	Tx22/Rx22	D6_0	Tx22/Rx22	D9_3	Tx22/Rx22
D2_7	Tx23/Rx23	D6_1	Tx23/Rx23	D9_4	Tx23/Rx23
LVAL	Tx24/Rx24	D6_2	Tx24/Rx24	D9_5	Tx24/Rx24
FVAL	Tx25/Rx25	D6_3	Tx25/Rx25	D9_6	Tx25/Rx25
D3_0	Tx26/Rx26	D6_4	Tx26/Rx26	D9_7	Tx26/Rx26
D3_1	Tx27/Rx27	LVAL	Tx27/Rx27	LVAL	Tx27/Rx27

X64-CL Sapera Capabilities

The three tables below describe the Sapera capabilities supported by the X64-CL Full and X64-CL Dual boards. Unless specified, each capability applies to both boards and all acquisition modes. Other Sapera capabilities have been omitted for clarity.

Specifically the X64-CL family is described in Sapera as:

- Board Server: X64_1
- Acquisition Device: Camera Link
- Acquisition Modes (X64-CL Full): Full Mono or Medium Color RGB
- Acquisition Modes (X64-CL Dual): Base Mono or Base Color RGB

Camera Related Capabilities

Capability	Values
CORACQ_CAP_CONNECTOR_TYPE	CORACQ_VAL_CONNECTOR_TYPE_CAMLINK (0x2)
CORACQ_CAP_CONNECTOR_CAMLINK (Pin – 01, Pin – 02, Pin – 03)	CORACQ_VAL_SIGNAL_NAME_NO_CONNECT (0x1) CORACQ_VAL_SIGNAL_NAME_PULSE0 (0x8) CORACQ_VAL_SIGNAL_NAME_PULSE1 (0x10) CORACQ_VAL_SIGNAL_NAME_GND (0x4000)
CORACQ_CAP_CONNECTOR_CAMLINK (Pin – 04)	CORACQ_VAL_SIGNAL_NAME_NO_CONNECT (0x1)

Table 2: Camera Related Capabilities

Camera Related Parameters

Parameter		Values
CORACQ_PRM_CHANNEL	*Medium/Full Mono*	CORACQ_VAL_CHANNEL_SINGLE (0x1) CORACQ_VAL_CHANNEL_DUAL (0x2)
	Base/Medium RGB / 10-taps	CORACQ_VAL_CHANNEL_SINGLE (0x1)
CORACQ_PRM_FRAME		CORACQ_VAL_FRAME_PROGRESSIVE (0x2)
CORACQ_PRM_INTERFACE		CORACQ_VAL_INTERFACE_DIGITAL (0x2)
CORACQ_PRM_SCAN		CORACQ_VAL_SCAN_AREA (0x1) CORACQ_VAL_SCAN_LINE (0x2)
CORACQ_PRM_SIGNAL		CORACQ_VAL_SIGNAL_DIFFERENTIAL (0x2)
CORACQ_PRM_VIDEO	*Mono/ RGB*	CORACQ_VAL_VIDEO_MONO (0x1) CORACQ_VAL_VIDEO_RGB (0x8)

CORACQ_PRM_PIXEL_DEPTH	*mono*	8 bits, # LUT = 1, LUT format = CORDATA_FORMAT_MONO8 10 bits, # LUT = 1, LUT format = CORDATA_FORMAT_MONO10 10 bits, # LUT = 1, LUT format = CORDATA_FORMAT_MONO8 14 bits, # LUT = 0, LUT format = CORDATA_FORMAT_MONO14 16 bits, # LUT = 0, LUT format = CORDATA_FORMAT_MONO16
	12-bit LUT	12 bits, # LUT = 1, LUT format = CORDATA_FORMAT_MONO12 12 bits, # LUT = 1, LUT format = CORDATA_FORMAT_MONO8
	color RGB	8 bits, # LUT = 1, LUT format = CORDATA_FORMAT_COLORNI8 10 bits, # LUT = 1, LUT format = CORDATA_FORMAT_COLORNI10
	12-bit LUT	12 bits, # LUT = 1, LUT format = CORDATA_FORMAT_COLORNI12
	10-taps	8 bits, # LUT = 1, LUT format = CORDATA_FORMAT_MONO8
CORACQ_PRM_VIDEO_STD		CORACQ_VAL_VIDEO_STD_NON_STD (0x1)
CORACQ_PRM_FIELD_ORDER		CORACQ_VAL_FIELD_ORDER_NEXT_FIELD (0x4)
CORACQ_PRM_HACTIVE	mono/color RGB	min = 1 pixel max = 16777215 pixel step = 1 pixel
	Bayer	min = 1 pixel max = 8192 pixel step = 1 pixel
CORACQ_PRM_HSYNC		min = 4 pixel max = 4294967295 pixel step = 1 pixel
CORACQ_PRM_VACTIVE		min = 1 line max = 16777215 line step = 1 line
CORACQ_PRM_VSYNC		min = 0 line max = 4294967295 line step = 1 line
CORACQ_PRM_HFRONT_INVALID		min = 0 pixel max = 16777215 pixel step = 1 pixel
CORACQ_PRM_HBACK_INVALID		min = 0 pixel max = 16777215 pixel step = 1 pixel
CORACQ_PRM_VFRONT_INVALID		min = 0 line max = 16777215 line step = 1 line
CORACQ_PRM_VBACK_INVALID		min = 0 line max = 16777215 line step = 1 line
CORACQ_PRM_PIXEL_CLK_SRC		CORACQ_VAL_PIXEL_CLK_SRC_EXT (0x2)
CORACQ_PRM_PIXEL_CLK_EXT		min = 20000000 Hz max = 85000000 Hz step = 1 Hz
CORACQ_PRM_SYNC		CORACQ_VAL_SYNC_SEP_SYNC (0x4)
CORACQ_PRM_HSYNC_POLARITY		CORACQ_VAL_ACTIVE_LOW (0x1)
CORACQ_PRM_VSYNC_POLARITY		CORACQ_VAL_ACTIVE_LOW (0x1)

CORACQ_PRM_TIME_INTEGRATE_METHOD		CORACQ_VAL_TIME_INTEGRATE_METHOD_1 (0x1) CORACQ_VAL_TIME_INTEGRATE_METHOD_2 (0x2) CORACQ_VAL_TIME_INTEGRATE_METHOD_3 (0x4) CORACQ_VAL_TIME_INTEGRATE_METHOD_4 (0x8) CORACQ_VAL_TIME_INTEGRATE_METHOD_5 (0x10) CORACQ_VAL_TIME_INTEGRATE_METHOD_6 (0x20) CORACQ_VAL_TIME_INTEGRATE_METHOD_7 (0x40) CORACQ_VAL_TIME_INTEGRATE_METHOD_8 (0x80) CORACQ_VAL_TIME_INTEGRATE_METHOD_9 (0x100)
CORACQ_PRM_CAM_TRIGGER_METHOD		CORACQ_VAL_CAM_TRIGGER_METHOD_1 (0x1) CORACQ_VAL_CAM_TRIGGER_METHOD_2 (0x2)
CORACQ_PRM_CAM_TRIGGER_POLARITY		CORACQ_VAL_ACTIVE_LOW (0x1) CORACQ_VAL_ACTIVE_HIGH (0x2)
CORACQ_PRM_CAM_TRIGGER_DURATION		min = 1 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_CAM_RESET_METHOD		CORACQ_VAL_CAM_RESET_METHOD_1 (0x1)
CORACQ_PRM_CAM_RESET_POLARITY		CORACQ_VAL_ACTIVE_LOW (0x1) CORACQ_VAL_ACTIVE_HIGH (0x2)
CORACQ_PRM_CAM_RESET_DURATION		min = 1 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_CAM_NAME	mono color RGB 10-taps	Default Area Scan 1 tap Mono Default Area Scan 1 tap Color Default Area Scan 10 taps Parallel Mono
CORACQ_PRM_LINE_INTEGRATE_METHOD		CORACQ_VAL_LINE_INTEGRATE_METHOD_1 (0x1) CORACQ_VAL_LINE_INTEGRATE_METHOD_2 (0x2) CORACQ_VAL_LINE_INTEGRATE_METHOD_3 (0x4) CORACQ_VAL_LINE_INTEGRATE_METHOD_4 (0x8) CORACQ_VAL_LINE_INTEGRATE_METHOD_7 (0x40)
CORACQ_PRM_LINE_TRIGGER_METHOD		CORACQ_VAL_LINE_TRIGGER_METHOD_1 (0x1)
CORACQ_PRM_LINE_TRIGGER_POLARITY		CORACQ_VAL_ACTIVE_LOW (0x1) CORACQ_VAL_ACTIVE_HIGH (0x2)
CORACQ_PRM_LINE_TRIGGER_DELAY		min = 0 µs max = 65535 µs step = 1 µs
CORACQ_PRM_LINE_TRIGGER_DURATION		min = 0 µs max = 65535 µs step = 1 µs
CORACQ_PRM_TAPS	*Full mono* *Base RGB* *Medium RGB* *10-taps*	min = 1 tap, max = 8 taps, step = 1 tap min = 1 tap, max = 1 tap, step = 1 tap min = 1 tap, max = 2 taps, step = 1 tap min = 10 taps, max = 10 taps, step = 1 tap
CORACQ_PRM_TAP_OUTPUT	Medium/Full mono color RGB 10-taps	CORACQ_VAL_TAP_OUTPUT_ALTERNATE (0x1) CORACQ_VAL_TAP_OUTPUT_SEGMENTED (0x2) CORACQ_VAL_TAP_OUTPUT_PARALLEL (0x4) CORACQ_VAL_TAP_OUTPUT_ALTERNATE (0x1) CORACQ_VAL_TAP_OUTPUT_SEGMENTED (0x2) CORACQ_VAL_TAP_OUTPUT_PARALLEL (0x4)
CORACQ_PRM_TAP_1_DIRECTION		CORACQ_VAL_TAP_DIRECTION_LR (0x1) CORACQ_VAL_TAP_DIRECTION_RL (0x2) CORACQ_VAL_TAP_DIRECTION_UD (0x4) CORACQ_VAL_TAP_DIRECTION_DU (0x8) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x10) CORACQ_VAL_TAP_DIRECTION_FROM_MID (0x20) CORACQ_VAL_TAP_DIRECTION_FROM_BOT (0x40)

CORACQ_PRM_TAP_2_DIRECTION		CORACQ_VAL_TAP_DIRECTION_LR (0x1) CORACQ_VAL_TAP_DIRECTION_RL (0x2) CORACQ_VAL_TAP_DIRECTION_UD (0x4) CORACQ_VAL_TAP_DIRECTION_DU (0x8) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x10) CORACQ_VAL_TAP_DIRECTION_FROM_MID (0x20) CORACQ_VAL_TAP_DIRECTION_FROM_BOT (0x40)
CORACQ_PRM_TAP_3_DIRECTION	*Full mono only*	CORACQ_VAL_TAP_DIRECTION_LR (0x1) CORACQ_VAL_TAP_DIRECTION_RL (0x2) CORACQ_VAL_TAP_DIRECTION_UD (0x4) CORACQ_VAL_TAP_DIRECTION_DU (0x8) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x10) CORACQ_VAL_TAP_DIRECTION_FROM_MID (0x20) CORACQ_VAL_TAP_DIRECTION_FROM_BOT (0x40)
CORACQ_PRM_TAP_4_DIRECTION	*Full mono only*	CORACQ_VAL_TAP_DIRECTION_LR (0x1) CORACQ_VAL_TAP_DIRECTION_RL (0x2) CORACQ_VAL_TAP_DIRECTION_UD (0x4) CORACQ_VAL_TAP_DIRECTION_DU (0x8) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x10) CORACQ_VAL_TAP_DIRECTION_FROM_MID (0x20) CORACQ_VAL_TAP_DIRECTION_FROM_BOT (0x40)
CORACQ_PRM_TAP_5_DIRECTION	*Full mono only*	CORACQ_VAL_TAP_DIRECTION_LR (0x1) CORACQ_VAL_TAP_DIRECTION_RL (0x2) CORACQ_VAL_TAP_DIRECTION_UD (0x4) CORACQ_VAL_TAP_DIRECTION_DU (0x8) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x10) CORACQ_VAL_TAP_DIRECTION_FROM_MID (0x20) CORACQ_VAL_TAP_DIRECTION_FROM_BOT (0x40)
CORACQ_PRM_TAP_6_DIRECTION	*Full mono only*	CORACQ_VAL_TAP_DIRECTION_LR (0x1) CORACQ_VAL_TAP_DIRECTION_RL (0x2) CORACQ_VAL_TAP_DIRECTION_UD (0x4) CORACQ_VAL_TAP_DIRECTION_DU (0x8) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x10) CORACQ_VAL_TAP_DIRECTION_FROM_MID (0x20) CORACQ_VAL_TAP_DIRECTION_FROM_BOT (0x40)
CORACQ_PRM_TAP_7_DIRECTION	*Full mono only*	CORACQ_VAL_TAP_DIRECTION_LR (0x1) CORACQ_VAL_TAP_DIRECTION_RL (0x2) CORACQ_VAL_TAP_DIRECTION_UD (0x4) CORACQ_VAL_TAP_DIRECTION_DU (0x8) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x10) CORACQ_VAL_TAP_DIRECTION_FROM_MID (0x20) CORACQ_VAL_TAP_DIRECTION_FROM_BOT (0x40)
CORACQ_PRM_TAP_8_DIRECTION	*Full mono only*	CORACQ_VAL_TAP_DIRECTION_LR (0x1) CORACQ_VAL_TAP_DIRECTION_RL (0x2) CORACQ_VAL_TAP_DIRECTION_UD (0x4) CORACQ_VAL_TAP_DIRECTION_DU (0x8) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x10) CORACQ_VAL_TAP_DIRECTION_FROM_MID (0x20) CORACQ_VAL_TAP_DIRECTION_FROM_BOT (0x40)
CORACQ_PRM_PIXEL_CLK_DETECTION		CORACQ_VAL_RISING_EDGE (0x4)
CORACQ_PRM_CHANNELS_ORDER		CORACQ_VAL_CHANNELS_ORDER_NORMAL (0x1) CORACQ_VAL_CHANNELS_ORDER_REVERSE (0x2)
CORACQ_PRM_CAM_LINE_TRIGGER_FREQ_MIN		1 Hz
CORACQ_PRM_CAM_LINE_TRIGGER_FREQ_MAX		16777215 Hz
CORACQ_PRM_CAM_TIME_INTEGRATE_DURATION_MIN		1 µs
CORACQ_PRM_CAM_TIME_INTEGRATE_DURATION_MAX		65535000 µs
CORACQ_PRM_TIME_INTEGRATE_PULSE1_POLARITY		CORACQ_VAL_ACTIVE_LOW (0x1) CORACQ_VAL_ACTIVE_HIGH (0x2)

CORACQ_PRM_TIME_INTEGRATE_PULSE1_DELAY		min = 0 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_TIME_INTEGRATE_PULSE1_DURATION		min = 0 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_CAM_IO_CONTROL (*)		
CORACQ_PRM_TIME_INTEGRATE_PULSE0_POLARITY		CORACQ_VAL_ACTIVE_LOW (0x1) CORACQ_VAL_ACTIVE_HIGH (0x2)
CORACQ_PRM_TIME_INTEGRATE_PULSE0_DELAY		min = 0 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_TIME_INTEGRATE_PULSE0_DURATION		min = 1 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_LINE_INTEGRATE_PULSE1_POLARITY		CORACQ_VAL_ACTIVE_LOW (0x1) CORACQ_VAL_ACTIVE_HIGH (0x2)
CORACQ_PRM_LINE_INTEGRATE_PULSE1_DELAY		min = 0 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_LINE_INTEGRATE_PULSE1_DURATION		min = 1 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_LINE_INTEGRATE_PULSE0_POLARITY		CORACQ_VAL_ACTIVE_LOW (0x1) CORACQ_VAL_ACTIVE_HIGH (0x2)
CORACQ_PRM_LINE_INTEGRATE_PULSE0_DELAY		min = 0 µs max = 65535 µs step = 1 µs
CORACQ_PRM_LINE_INTEGRATE_PULSE0_DURATION		min = 1 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_CAMLINK_ CONFIGURATION	Full Mono	CORACQ_VAL_CAMLINK_CONFIGURATION_BASE (0x1) CORACQ_VAL_CAMLINK_CONFIGURATION_MEDIUM (0x2) CORACQ_VAL_CAMLINK_CONFIGURATION_FULL (0x4) CORACQ_VAL_CAMLINK_CONFIGURATION_2BASE (0x8)
	medium color RGB	CORACQ_VAL_CAMLINK_CONFIGURATION_BASE (0x1) CORACQ_VAL_CAMLINK_CONFIGURATION_MEDIUM (0x2)
	10-taps	CORACQ_VAL_CAMLINK_CONFIGURATION_10TAPS_FORMAT2 (0x40)
CORACQ_PRM_DATA_VALID_ENABLE	Mono	TRUE FALSE
	10 taps	Not available
CORACQ_PRM_DATA_VALID_POLARITY		CORACQ_VAL_ACTIVE_HIGH (0x2)
CORACQ_PRM_TAP_9_DIRECTION	*10-taps only*	CORACQ_VAL_TAP_DIRECTION_LR (0x1) CORACQ_VAL_TAP_DIRECTION_RL (0x2) CORACQ_VAL_TAP_DIRECTION_UD (0x4) CORACQ_VAL_TAP_DIRECTION_DU (0x8) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x10) CORACQ_VAL_TAP_DIRECTION_FROM_MID (0x20) CORACQ_VAL_TAP_DIRECTION_FROM_BOT (0x40)

CORACQ_PRM_TAP_10_DIRECTION	*10-taps only*	CORACQ_VAL_TAP_DIRECTION_LR (0x1) CORACQ_VAL_TAP_DIRECTION_RL (0x2) CORACQ_VAL_TAP_DIRECTION_UD (0x4) CORACQ_VAL_TAP_DIRECTION_DU (0x8) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x10) CORACQ_VAL_TAP_DIRECTION_FROM_MID (0x20) CORACQ_VAL_TAP_DIRECTION_FROM_BOT (0x40)
CORACQ_PRM_TIMESLOT	10-taps	CORACQ_VAL_TIMESLOT_1 (0x1)
CORACQ_PRM_CAM_CONTROL_DURING_READOUT		TRUE FALSE
CORACQ_PRM_LINE_TRIGGER_AUTO_DELAY		Not available

Table 3: Camera Related Parameters

VIC Related Parameters

Parameter		Values
CORACQ_PRM_CAMSEL	Mono	CAMSEL_MONO = from 0 to 0
	color RGB	CAMSEL_RGB = from 0 to 0
CORACQ_PRM_CROP_LEFT		min = 0 pixel max = 16777215 pixel step = 8 pixel
CORACQ_PRM_CROP_TOP		min = 0 line max = 16777215 line step = 1 line
CORACQ_PRM_CROP_WIDTH		min = 8 pixel max = 16777215 pixel step = 8 pixel step = 2 with RGB
CORACQ_PRM_CROP_HEIGHT		min = 1 line max = 16777215 line step = 1 line
CORACQ_PRM_DECIMATE_METHOD		CORACQ_VAL_DECIMATE_DISABLE (0x1)
CORACQ_PRM_LUT_ENABLE		TRUE FALSE (not available with 10-tap)
CORACQ_PRM_LUT_NUMBER		Default = 0
CORACQ_PRM_STROBE_ENABLE		TRUE FALSE
CORACQ_PRM_STROBE_METHOD		CORACQ_VAL_STROBE_METHOD_1 (0x1) CORACQ_VAL_STROBE_METHOD_2 (0x2) CORACQ_VAL_STROBE_METHOD_3 (0x4) CORACQ_VAL_STROBE_METHOD_4 (0x8)
CORACQ_PRM_STROBE_POLARITY		CORACQ_VAL_ACTIVE_LOW (0x1) CORACQ_VAL_ACTIVE_HIGH (0x2)
CORACQ_PRM_STROBE_DURATION		min = 0 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_STROBE_DELAY		min = 0 µs max = 65535000 µs step = 1 µs

CORACQ_PRM_TIME_INTEGRATE_ENABLE		TRUE FALSE
CORACQ_PRM_TIME_INTEGRATE_DURATION		min = 1 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_CAM_TRIGGER_ENABLE		TRUE FALSE
CORACQ_PRM_CAM_RESET_ENABLE		TRUE FALSE
CORACQ_PRM_OUTPUT_FORMAT	*mono*	CORACQ_VAL_OUTPUT_FORMAT_MONO8 CORACQ_VAL_OUTPUT_FORMAT_MONO16
	color RGB	CORACQ_VAL_OUTPUT_FORMAT_RGB8888 CORACQ_VAL_OUTPUT_FORMAT_RGB101010
CORACQ_PRM_EXT_TRIGGER_ENABLE		CORACQ_VAL_EXT_TRIGGER_OFF (0x1) CORACQ_VAL_EXT_TRIGGER_ON (0x8)
CORACQ_PRM_VIC_NAME	Mono	Default Area Scan 1 tap Mono
	Color RGB	Default Area Scan 1 tap Color
	10-taps	Default Area Scan 10 taps Parallel Mono
CORACQ_PRM_LUT_MAX		1 0 (with 10-tap)
CORACQ_PRM_EXT_TRIGGER_DETECTION		CORACQ_VAL_ACTIVE_LOW (0x1) CORACQ_VAL_ACTIVE_HIGH (0x2) CORACQ_VAL_RISING_EDGE (0x4) CORACQ_VAL_FALLING_EDGE (0x8)
CORACQ_PRM_LUT_FORMAT	*mono*	Default = CORACQ_VAL_OUTPUT_FORMAT_MONO8
	color RGB	Default = CORACQ_VAL_OUTPUT_FORMAT_RGB101010
	10-taps	Not Available
CORACQ_PRM_VSYNC_REF		CORACQ_VAL_SYNC_REF_END (0x2)
CORACQ_PRM_HSYNC_REF		CORACQ_VAL_SYNC_REF_END (0x2)
CORACQ_PRM_LINE_INTEGRATE_ENABLE		TRUE FALSE
CORACQ_PRM_LINE_INTEGRATE_DURATION		min = 1 pixel max = 16777215 pixel step = 1 pixel
CORACQ_PRM_LINE_TRIGGER_ENABLE		TRUE FALSE
CORACQ_PRM_EXT_FRAME_TRIGGER_ENABLE		TRUE FALSE
CORACQ_PRM_EXT_FRAME_TRIGGER_DETECTION		CORACQ_VAL_ACTIVE_LOW (0x1) CORACQ_VAL_ACTIVE_HIGH (0x2) CORACQ_VAL_RISING_EDGE (0x4) CORACQ_VAL_FALLING_EDGE (0x8) CORACQ_VAL_DOUBLE_PULSE_RISING_EDGE (0x20) CORACQ_VAL_DOUBLE_PULSE_FALLING_EDGE (0x40)
CORACQ_PRM_EXT_LINE_TRIGGER_ENABLE		TRUE FALSE
CORACQ_PRM_EXT_LINE_TRIGGER_DETECTION		CORACQ_VAL_RISING_EDGE (0x4)
CORACQ_PRM_SNAP_COUNT		min = 1 frame max = 65535 frame step = 1 frame
CORACQ_PRM_INT_LINE_TRIGGER_ENABLE		TRUE FALSE

CORACQ_PRM_INT_LINE_TRIGGER_FREQ		Default = 5000 Hz
CORACQ_PRM_BIT_ORDERING		CORACQ_VAL_BIT_ORDERING_STD (0x1)
CORACQ_PRM_EXT_TRIGGER_LEVEL		CORACQ_VAL_LEVEL_TTL (0x1) CORACQ_VAL_LEVEL_422 (0x2)
CORACQ_PRM_STROBE_LEVEL		CORACQ_VAL_LEVEL_TTL (0x1)
CORACQ_PRM_EXT_FRAME_TRIGGER_LEVEL		CORACQ_VAL_LEVEL_TTL (0x1) CORACQ_VAL_LEVEL_422 (0x2)
CORACQ_PRM_EXT_LINE_TRIGGER_LEVEL		CORACQ_VAL_LEVEL_422 (0x2)
CORACQ_PRM_INT_LINE_TRIGGER_FREQ_MIN		245 Hz
CORACQ_PRM_INT_LINE_TRIGGER_FREQ_MAX		500000 Hz
CORACQ_PRM_MASTER_MODE		Not available
CORACQ_PRM_SHAFT_ENCODER_DROP		min = 0 tick max = 255 tick step = 1 tick
CORACQ_PRM_SHAFT_ENCODER_ENABLE		TRUE FALSE
CORACQ_PRM_EXT_TRIGGER_FRAME_COUNT		min = 1 frame max = 65534 frame step = 1 frame
CORACQ_PRM_INT_FRAME_TRIGGER_ENABLE		TRUE FALSE
CORACQ_PRM_INT_FRAME_TRIGGER_FREQ		min = 1 milli-Hz max = 1073741823 milli-Hz step = 1 milli-Hz
CORACQ_PRM_STROBE_DELAY_2		min = 0 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_FRAME_LENGTH		CORACQ_VAL_FRAME_LENGTH_FIX (0x1) CORACQ_VAL_FRAME_LENGTH_VARIABLE (0x2)
CORACQ_PRM_FLIP	mono	CORACQ_VAL_FLIP_OFF (0x00) CORACQ_VAL_FLIP_HORZ (0x01)
	10 taps	Not available
CORACQ_PRM_EXT_TRIGGER_DURATION		min = 0 µs max = 255 µs step = 1 µs
CORACQ_PRM_TIME_INTEGRATE_DELAY		min = 0 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_CAM_RESET_DELAY		min = 0 µs max = 0 µs step = 1 µs
CORACQ_PRM_CAM_TRIGGER_DELAY		min = 0 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_SHAFT_ENCODER_LEVEL		CORACQ_VAL_LEVEL_422 (0x2)
CORACQ_PRM_EXT_FRAME_TRIGGER_SOURCE (*)		min = 0 max = 5 step = 1
CORACQ_PRM_EXT_LINE_TRIGGER_SOURCE (*)		min = 0 max = 7 step = 1

CORACQ_PRM_EXT_TRIGGER_SOURCE (*)	min = 0 max = 5 step = 1
CORACQ_PRM_SHAFT_ENCODER_MULTIPLY	min = 1 max = 32 step = (2**N)
CORACQ_PRM_PLANAR_INPUT_SOURCES	Not available
CORACQ_PRM_EXT_TRIGGER_DELAY	Not available
CORACQ_PRM_BAYER_DECODER_METHOD	Not available
CORACQ_PRM_EXT_TRIGGER_IGNORE_DELAY	min = 0 max = 65535000 step = 1
CORACQ_PRM_EXT_TRIGGER_SOURCE_STR	[0] = Automatic [1] = From External Trigger #1 [2] = From External Trigger #2 [3] = From Board Sync [4] = To Board Sync [5] = Pulse to Board Sync
CORACQ_PRM_EXT_LINE_TRIGGER_SOURCE_STR	[0] = Automatic [1] = From Shaft Encoder Phase A [2] = From Shaft Encoder Phase B [3] = From Shaft Encoder Phase A & B [4] = From Board Sync [5] = To Board Sync [6] = Pulse to Board Sync [7] = To Board Sync When Grabbing
CORACQ_PRM_VERTICAL_TIMEOUT_DELAY	min = 0 max = 16383000 step = 1
CORACQ_PRM_POCL_ENABLE	Not available
CORACQ_PRM_SHAFT_ENCODER_SOURCE (*)	Not available
CORACQ_PRM_SHAFT_ENCODER_DIRECTION	Not available

Table 4: VIC Related Parameters

ACQ Related Parameters

Parameter		Values
CORACQ_PRM_LABEL	Full mono color RGB 10 taps	Camera Link Full Mono #1 Camera Link Medium Color RGB #1 Camera Link 10 Taps Parallel Mono #1
	Dual mono/RGB	Camera Link Base Mono #1 Camera Link Base Mono #2 CameraLink Base Color RGB #1 CameraLink Base Color RGB #2
CORACQ_PRM_EVENT_TYPE		CORACQ_VAL_EVENT_TYPE_EXTERNAL_TRIGGER CORACQ_VAL_EVENT_TYPE_NO_PIXEL_CLK CORACQ_VAL_EVENT_TYPE_PIXEL_CLK CORACQ_VAL_EVENT_TYPE_FRAME_LOST CORACQ_VAL_EVENT_TYPE_DATA_OVERFLOW CORACQ_VAL_EVENT_TYPE_EXTERNAL_TRIGGER_IGNORED
CORACQ_PRM_SIGNAL_STATUS		CORACQ_VAL_SIGNAL_HSYNC_PRESENT CORACQ_VAL_SIGNAL_VSYNC_PRESENT CORACQ_VAL_SIGNAL_PIXEL_CLK_PRESENT
CORACQ_PRM_FLAT_FIELD_ ENABLE		Not available
CORACQ_CAP_SERIAL_PORT_INDEX		Supported

Table 5: Acquisition Related Parameters

X64-CL Sapera Servers & Resources

Servers and Resources

The following table describes the X64-CL Full board

Servers		Resources		
Name	**Type**	**Name**	**Index**	**Description**
X64_1 (default firmware & 12-bit LUT)	Acquisition	Camera Link Full Mono #1	0	Full configuration, monochrome output, Camera #1
		Camera Link Medium Color RGB #1	1	Medium configuration, RGB output, Camera #1
X64_1 (10 taps firmware)	Acquisition	Camera Link 10 Taps Parallel Mono #1	0	Full configuration, 10 Taps @ 8 bits, Camera #1

Table 6: X64-CL Full Board - Servers and Resources

The following table describes the X64-CL Dual board

Servers		Resources		
Name	**Type**	**Name**	**Index**	**Description**
X64_1 (default Base firmware)	Acquisition	Camera Link Base Mono 1	0	Base configuration, monochrome Camera #1
		Camera Link Base Mono 2	1	Base configuration, monochrome Camera #2
		Camera Link Base RGB 1	2	Base configuration, color RGB Camera #1
		Camera Link Base RGB 2	3	Base configuration, color RGB Camera #2

Table 7: X64-CL Dual Board - Servers and Resources

Technical Specifications

Board Specifications

Number	Dependent on Model X64-CL Full: 1 Full or 1 Medium or 1 Base X64-CL Dual: 1 or 2 Base
Common Pixel Formats *(see details following)*	Monochrome cameras - 8, 10, or 12 bits RGB cameras – 24 bits
Base Configuration Pixel Formats	3 x 8-bit, 2 x 10-bit, 2 x 12-bit, 1 x 14-bit, 1 x 16-bit, 24-bit RGB
Medium Configuration Pixel Formats	4 x 8-bit, 2 x 10-bit, 4 x 10-bit, 2 x 12-bit, 4 x 12-bit, 30-bit RGB
Full Configuration Pixel Formats	8 x 8-bit, 10 x 8-bit (non-standard)
Scanning	Progressive, Multi-Tap, Multi-Channel, Four quadrant, Tap reversal
Input LUT	Yes. See section "Output LUT Availability" (page 64) for details.
Resolution *note: these are X64-CL maximums, not Camera Link specifications*	Horizontal Minimum: 8 Pixels per tap (X64-CL) 16 Pixels per tap (X64-EM) Horizontal Maximum: 8-bits/pixel x 256K Pixels/line 16-bits/pixel x 128K Pixels/line 32-bits/pixel x 64K Pixels/line 64-bits/pixel x 32K Pixels/line Vertical Minimum: 1 line Vertical Maximum: up to 16,000,000 lines
Pixel Clock Range	up to 85 MHz dependent on model
Controls	Compliant with Teledyne DALSA Trigger-to-Image Reliability framework Comprehensive event notification Timing control logic for EXSYNC, PRIN and strobe signals Dual independent TTL/LVDS trigger input programmable as active high or low (edge or level trigger) (minimum trigger pulse = 100ns) Dual independent TTL Strobe outputs Quadrature (AB) shaft-encoder inputs for external web synchronization
Serial Port	Supported communication speeds: 9600 to 115 kbps

Supported Specialty Cameras

Contact Teledyne DALSA support (http://www.imaging.com/support) for more information on these cameras and other similar non-standard cameras.

See "Firmware: X64-CL 10-Tap Format" on page 65 for more information about the X64-CL support for the Basler A504k.

X64-CL Physical Dimensions

Approximately 6.9 in. (17.5 cm) wide by 4.2 in. (10.7 cm) high (conforms to half length PCI)

Host System Requirements

General System Requirements for the X64-CL Series

Computer system with a 64 bit – 66/33 MHz PCI slot or a 32 bit – 33 MHz PCI slot.

Operating System Support

Windows XP, Windows Vista and Windows 7, either 32-bit or 64-bit

Power Requirements

+5 Volt	2 amp typical	*Note: other internal voltages are derived from +5V*
+12 Volt		*As per camera connected and supplied by X64-CL PC power interface*

Environment

Ambient Temperature:	10° to 50° C (operation) 0° to 70° C (storage)
Relative Humidity:	5% to 90% non-condensing (operating) 0% to 95% (storage)

EMI Certifications

Class B, both FCC and CE.

EC & FCC DECLARATION OF CONFORMITY

We : CORECO INC.
7075 Place Robert-Joncas, Suite 142,
St. Laurent, Quebec, Canada H4M 2Z2

Declare under sole legal responsibility that the following products conform to the protection requirements of council directive 89/336 EEC on the approximation of the laws of member states relating to electromagnetic compatibility, as amended by directive 93/68/EEC :

FRAME GRABBER BOARD: X64-CL

The products to which this declaration relates are in conformity with the following relevant harmonised standards, the reference numbers of which have been published in the Official Journal of the European Communities :

EN55022 :1998- Residential, Commercial and Light Industry
EN50204: 1995
EN61000-4: 1995, 1996

Further declare under our sole legal responsibility that the product listed conforms to the code of federal regulations CFR 47 part 15 for a class B product.

St. Laurent, Canada — Location
May 13, 2003 — Date

Ghislain Beaupré
Vice-President,
Research & Development

State-of-the-art imaging products

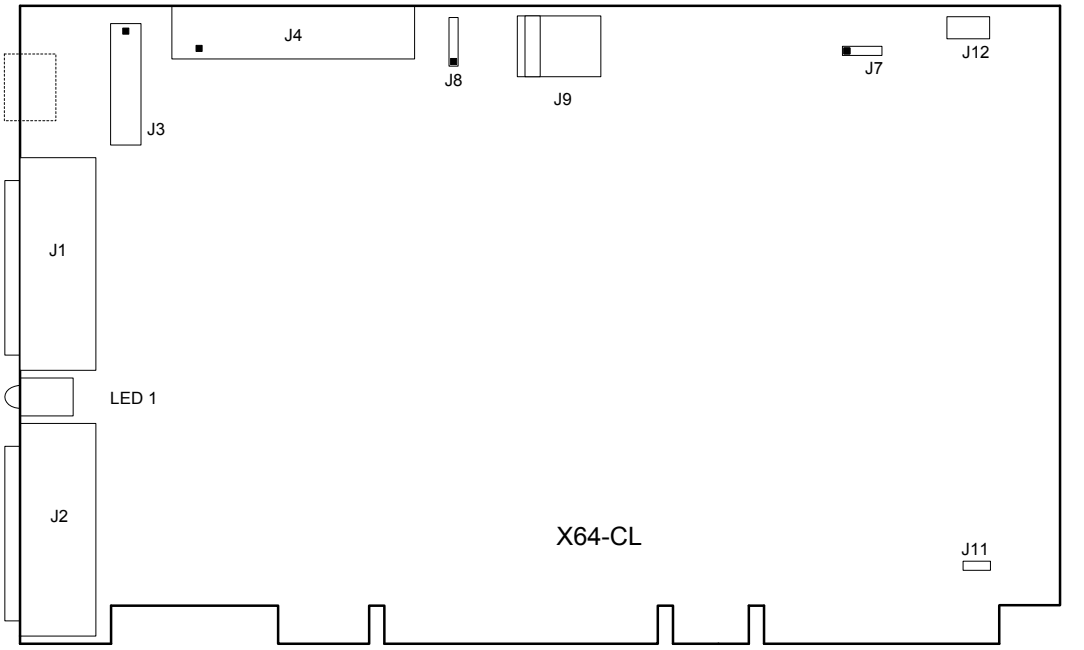
Coreco Imaging 7075 Place Robert-Joncas, Suite 142, Saint-Laurent, Quebec, Canada H4M 2Z2
Telephone: (514) 333-1301 Fax: (514) 333-1388 www.imaging.com

Connector and Switch Locations

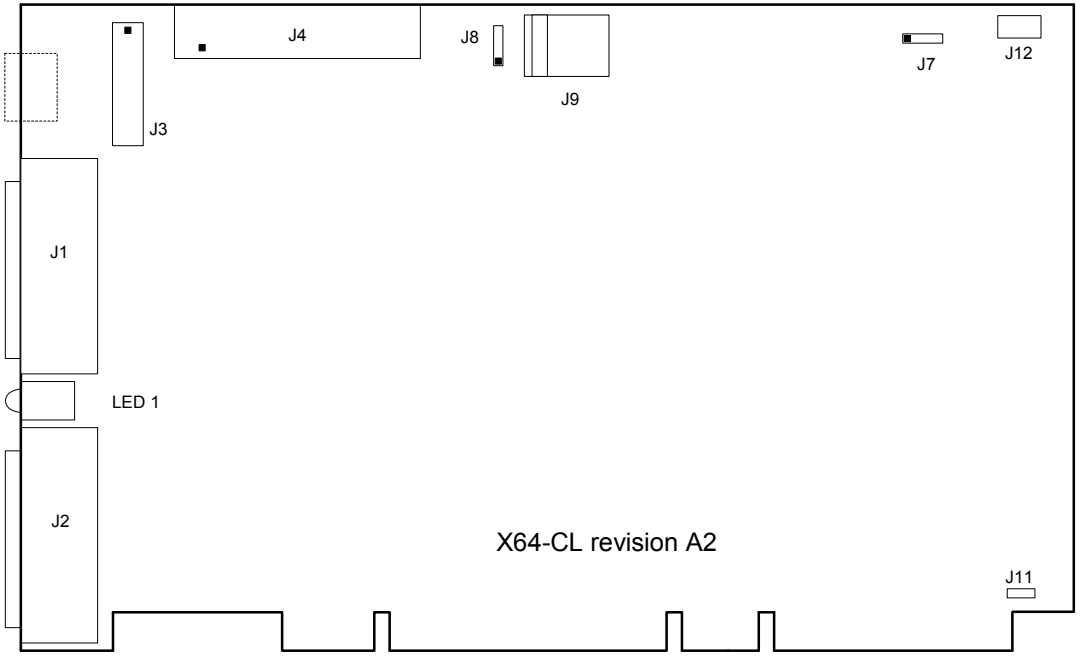
X64-CL Layout Drawings

The X64-CL is built on a half length board. The component view drawings are shown below.

X64-CL Revision A3, A4, A5, A6, A7, A8 Layout Drawing



X64-CL Revision A2 Layout Drawing

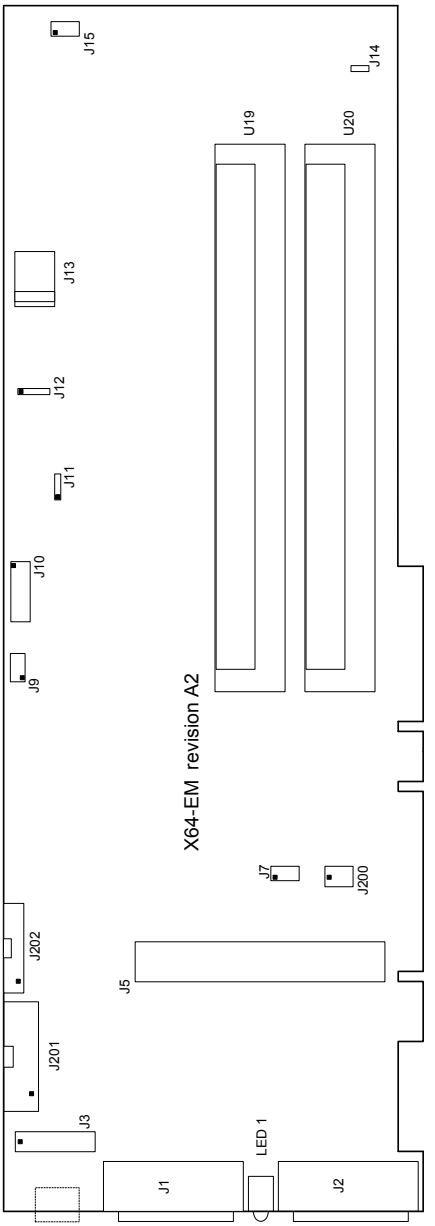


Connector List (X64-CL half length board)

Connector	Description	Connector	Description
J1	Camera Link Connector	J8	Camera Power Selector
J2	Camera Link Connector	J9	PC power to camera interface.
J3	Aux. Module Connector (typical – Hirose)	J11	Normal (jumper on) Safe Start Mode (jumper off)
J4	External Signals Connector block	J12	Reserved
J7	Multiple board trigger lock		

X64-CL EM Revision A2 Layout Drawing

The X64-CL Series boards with 256MB, 1GB, or 2GB of memory, is built on a full length board. For reference purposes, this PCB is called X64-CL EM. The board layout and connector list are shown below.



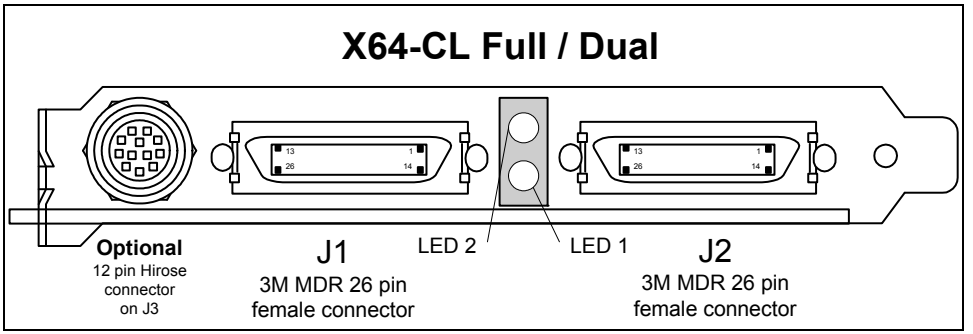
Connector List (X64-CL EM full length board)

Connector	Description	Connector	Description
J1	Camera Link Connector	J11	Multiple board trigger lock
J2	Camera Link Connector	J12	Camera Power Selector
J3	Aux. Module Connector (typical – Hirose)	J13	PC power to camera interface.
J5	Connector for Camera Link Expansion module	J14	Normal (jumper on) Safe Start Mode (jumper off)
J201	External Signals Connector block	J7, J9, J10, J15, J200	Reserved
J202	Reserved	U19, U20	DDR memory (registered) modules

Connector and Switch Specifications

X64-CL Connector View

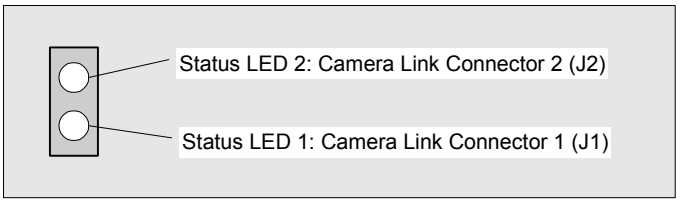
The following X64-CL connector bracket view shows the Camera Link connectors, status LEDs and the optional 12 pin Hirose connector module (connected internally to connector J3). Note that some computer cases may not provide the required clearance for the Hirose connector option.



- The **X64-CL Full** board supports a camera with one or two Camera Link MDR-26 connectors (Base or medium or Full – see "Data Port Summary" on page 110 for information on Camera Link configurations).
    - Connect the camera to the X64-CL J1 connector with a Camera Link cable. When using Medium or Full cameras, connect the second camera connector to X64-CL J2.
- The **X64-CL Dual** supports one or two Base Camera Link cameras.
    - Connect the first camera to the X64-CL J1 connector with a Camera Link cable. If using a second camera, connect to X64-CL J2.

Caution: If the camera is powered by the X64-CL via the Hirose connector or External Signals Connector block, it is very important that the correct power supply voltage is selected correctly. Refer to "J8 (CL), J12 (CL EM): Power to Camera Voltage Selector" on page 105 for information on the selection jumper.

Status LEDs Functional Description



Status LED Modes

- Red: No camera connected or camera has no power.
- Green: Camera connected and is ON. Camera clock detected. No line valid detected.
- Slow Flashing Green: Camera Line Valid signal detected.
- Fast Flashing Green: Acquisition in progress.
- Status LED 2 flashing red, X64-CL Full board only: Camera pixel clock incorrectly connected to J2 instead of J1. (Example - a Base camera is incorrectly connected to J2).

J1: Camera Link Connector 1 (applies to X64-CL Dual & Full models)

Name	Pin #	Type	Description
BASE_X0-	25	Input	Neg. Base Data 0
BASE_X0+	12	Input	Pos. Base Data 0
BASE_X1-	24	Input	Neg. Base Data 1
BASE_X1+	11	Input	Pos. Base Data 1
BASE_X2-	23	Input	Neg. Base Data 2
BASE_X2+	10	Input	Pos. Base Data 2
BASE_X3-	21	Input	Neg. Base Data 3
BASE_X3+	8	Input	Pos. Base Data 3
BASE_XCLK-	22	Input	Neg. Base Clock
BASE_XCLK+	9	Input	Pos. Base Clock
SERTC+	20	Output	Pos. Serial Data to Camera
SERTC-	7	Output	Neg. Serial Data to Camera
SERTFG-	19	Input	Neg. Serial Data to Frame Grabber
SERTFG+	6	Input	Pos. Serial Data to Frame Grabber
CC1-	18	Output	Neg. Camera Control 1
CC1+	5	Output	Pos. Camera Control 1
CC2+	17	Output	Pos. Camera Control 2
CC2-	4	Output	Neg. Camera Control 2
CC3-	16	Output	Neg. Camera Control 3
CC3+	3	Output	Pos. Camera Control 3
CC4+	15	Output	Pos. Camera Control 4
CC4-	2	Output	Neg. Camera Control 4
GND	1, 13, 14, 26		Ground

J2: Camera Link Connector 2 (on X64-CL Dual model)

The Camera Link connector J2 on the X64-CL Dual board is identical to Camera Link connector 1 (J1).

J2: Camera Link Connector 2 (on X64-CL Full model used with Medium or Full cameras)

Name	Pin #	Type	Description
MEDIUM _X0-	25	Input	Neg. Medium Data 0
MEDIUM _X0+	12	Input	Pos. Medium Data 0
MEDIUM _X1-	24	Input	Neg. Medium Data 1
MEDIUM _X1+	11	Input	Pos. Medium Data 1
MEDIUM _X2-	23	Input	Neg. Medium Data 2
MEDIUM _X2+	10	Input	Pos. Medium Data 2
MEDIUM _X3-	21	Input	Neg. Medium Data 3
MEDIUM _X3+	8	Input	Pos. Medium Data 3
MEDIUM _XCLK-	22	Input	Neg. Medium Clock
MEDIUM _XCLK+	9	Input	Pos. Medium Clock
TERM	20		Term Resistor
TERM	7		Term Resistor
FULL_X0-	19	Input	Neg. Full Data 0
FULL _X0+	6	Input	Pos. Full Data 0
FULL _X1-	18	Input	Neg. Full Data 1
FULL _X1+	5	Input	Pos. Full Data 1
FULL _X2-	17	Input	Neg. Full Data 2
FULL _X2+	4	Input	Pos. Full Data 2
FULL _X3-	15	Input	Neg. Full Data 3
FULL _X3+	2	Input	Pos. Full Data 3
FULL _XCLK-	16	Input	Neg. Full Clock
FULL _XCLK+	3	Input	Pos. Full Clock
GND	1, 13, 14, 26		Ground

Camera Link Camera Control Signal Overview

Four LVDS pairs are for general-purpose camera control, defined as camera inputs / frame grabber outputs by the Camera Link Base camera specification. These controls are on J1 (X64-CL Full and X64-CL Dual) and also on J2 for the second Base camera input of the X64-CL Dual. The X64-CL can assign any camera control signal to the appropriate Camera Link control.

- Camera Control 1 (CC1)
- Camera Control 2 (CC2)
- Camera Control 3 (CC3)
- Camera Control 4 (CC4)

Each camera manufacture is free to define the signals input on any one or all four control signals. These control signals are used either as camera control pulses or as a static logic state. Control signals not required by the camera are simply assigned as not used. Refer to your camera's user manual for information on what control signals are required.

Note: The X64-CL pulse controller has a minimum resolution of 1 us. When configuring the Camera Link control signals, such as exposure control, etc. use values in increments of 1 us.

J4 (CL or EM rev. A0, A1): External Signals Connector

Warning: Proceed with caution when connecting external devices or other computers to the signals available on J4. Grounds should connect first and devices should be power up at the same time. External signal sources must not have voltage spikes or transients, else damage may occur on the X64-CL.

J4 Pin Header Numbering Detail

2	4	...	24	26
1	3	...	23	25

J4 Signal Descriptions

Pin #	Signal	X64-CL Full Description	X64-CL Dual Description
1		Reserved	Reserved
2, 4, 6	GND		
3		Reserved	Reserved
5		Reserved	Reserved
7		Reserved	Reserved
8		Reserved	Reserved
9		Reserved	Reserved
10		Reserved	Reserved
11	TrigIn 1 + (*input*)	LVDS Trigger In + or TTL Trigger In (*see note 1*)	*CamLink Base #1* LVDS + or TTL Trigger In (*see note 2*)
12	TrigIn 1 - (*input*)	LVDS Trigger In - (or TTL Trigger In GND)	*CamLink Base #1* LVDS Trigger - (or TTL Trigger GND)
13	TrigIn 2 + (*input*)	LVDS Trigger In + or TTL Trigger In (*Used for two pulse external trigger with variable frame length linescan acquisition*)	*CamLink Base #2* LVDS + or TTL Trigger In (*see note 2*)
14	TrigIn 2 - (*input*)	LVDS Trigger In - (or TTL Trigger In GND)	*CamLink Base #2* LVDS Trigger - (or TTL Trigger GND)
15	Phase A + (*input*)	LVDS Shaft Encoder phase A + or line trigger used with linescan cameras (*see note 3 & 4 & 9*)	LVDS Shaft Encoder + or line trigger used with linescan cameras (*see note 5 & 9*)
16	Phase A - (*input*)	LVDS Shaft Encoder phase A - (*see note 9*)	LVDS Shaft Encoder - (*see note 9*)
17	Phase B + (*input*)	LVDS Shaft Encoder phase B + or line trigger used with linescan cameras (*see note 3 & 4 & 9*)	LVDS Shaft Encoder + or line trigger used with linescan cameras (*see note 5 & 9*)
18	Phase B - (*input*)	LVDS Shaft Encoder phase B - (*see note 9*)	LVDS Shaft Encoder - (*see note 9*)
19	Strobe 2 (*output*)	*not used*	*CamLink Base #2* TTL Strobe Output (*see note 7*)
20		Reserved	

21	Strobe 1 (*output*)	TTL Strobe Output (*see note 6*)	*CamLink Base #1* TTL Strobe Output (*see note 7*)
22, 24, 26	GND		
23, 25	DC Power (*see note 8*)	Voltage selected (+12 or +5) via J8 (see "J8 (CL), J12 (CL EM): Power to Camera Voltage Selector" on page 105)	Voltage selected (+12 or +5) via J8 (see "J8 (CL), J12 (CL EM): Power to Camera Voltage Selector" on page 105)

Notes for X64-CL Full or X64-CL Dual External Signals Connector

1. **X64-CL Full:**
   Refer to Sapera parameters CORACQ_PRM_EXT_TRIGGER_LEVEL
   CORACQ_PRM_EXT_FRAME_TRIGGER_LEVEL
   CORACQ_PRM_EXT_TRIGGER_ENABLE
   CORACQ_PRM_EXT_TRIGGER_DETECTION
   See also *.cvi file entries:
   External Trigger Level, External Frame Trigger Level, External Trigger Enable, External Trigger Detection.

2. **X64-CL Dual:**
   Sapera parameter CORACQ_PRM_EXT_TRIGGER_LEVEL, CORACQ_PRM_EXT_FRAME_TRIGGER_LEVEL is a common control to both Camera Link Base #1 and #2.
   Parameters CORACQ_PRM_EXT_TRIGGER_ENABLE and CORACQ_PRM_EXT_TRIGGER_DETECTION are independent for each Camera Link input.
   When detection is CORACQ_VAL_DOUBLE _PULSE_RISING_EDGE or CORACQ_VAL_DOUBLE _PULSE_FALLING_EDGE the start trigger is "Trig In 1" and the end trigger is "Trig in 2".
   See also *.cvi file entries:
   External Trigger Level, External Frame Trigger Level, External Trigger Enable, External Trigger Detection.

3. **X64-CL Full:**
   See "Line Trigger Source Selection for Linescan Applications" on page 56 for more information.
   Refer to Sapera parameters CORACQ_PRM_SHAFT_ENCODER_ENABLE
   CORACQ_PRM_SHAFT_ENCODER_DROP
   or refer to CORACQ_PRM_EXT_LINE_TRIGGER_ENABLE
   CORACQ_PRM_EXT_LINE_TRIGGER_DETECTION
   CORACQ_PRM_EXT_LINE_TRIGGER_LEVEL (fixed at LVDS)
   CORACQ_PRM_EXT_LINE_TRIGGER_SOURCE
   See also *.cvi file entries:
   Shaft Encoder Enable, Shaft Encoder Pulse Drop
   or see External Line Trigger Enable, External Line Trigger Detection, External Line Trigger Level, External Line Trigger Source.

4. **X64-CL Full:**
   Important: When using only one shaft encoder input phase, say phase A, then the phase B inputs must be terminated by connecting phase B- to board ground available on any pin labeled GND and phase B+ to any DC source with a minimum of 100 mV positive relative to the phase B- input.

5. **X64-CL Dual:**
   Same parameters as X64-CL Full (see note 3).
   Parameters are independent for Camera Link Base #1 and #2.

6. **X64-CL Full:**
   Refer to Sapera parameters CORACQ_PRM_STROBE_ENABLE CORACQ_PRM_STROBE_POLARITY, CORACQ_PRM_STROBE_LEVEL, CORACQ_PRM_STROBE_METHOD, CORACQ_PRM_STROBE_DELAY CORACQ_PRM_STROBE_DURATION
   See also *.cvi file entries:
   Strobe Enable, Strobe Polarity, Strobe Level, Strobe Method, Strobe Delay, Strobe Duration.

7. **X64-CL Dual:**
   Same parameters as X64-CL Full (see note 6).
   Parameters are independent for Camera Link Base #1 and #2.

8. **DC Power Constraints:**
   The supplied host PC voltage is selected (+12 or +5) via the shorting jumper J8 (see "J8 (CL), J12 (CL EM): Power to Camera Voltage Selector" on page 105 for details).
   Important.
   For **X64-CL revision A2, A3, A4** boards, the selected voltage is *not* fused. It is strongly recommended that any cabling using the DC power has an inline fuse (1.5A max).
   For **X64-CL revision A5 and later** boards, a 1.5A resettable fuse is included on the board. If the fuse is tripped, power off the host computer to reset the fuse.

9. See "Connecting a TTL Shaft Encoder Signal to the LVDS Input" on page 101 for details on using a TTL shaft encoder signal.

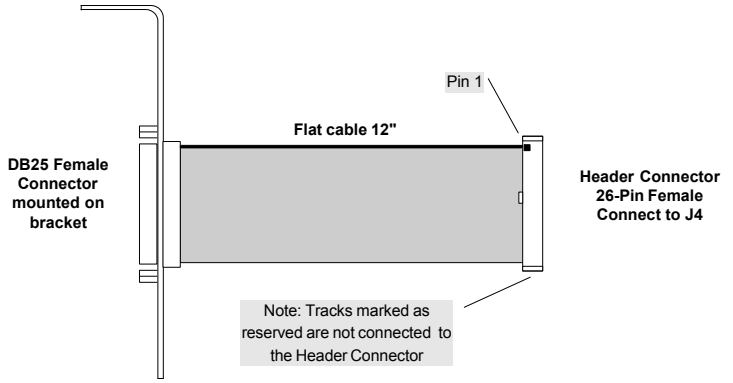
X64-CL or X64-EM rev. A0, A1: External Signals Connector Bracket Assembly

The External Signals Connector Bracket Assembly (part number **OR-64CC-0TIO2**) applies to the short length X64-CL board with 32 or 128 MB of memory. For the X64-CL EM full length board with 256MB, 1GB, or 2GB see “X64-EM rev. A2 & later: External Signals Connector Bracket Assembly” on page 97.

The External Signals connector bracket provides a simple way to bring out the signals from the X64-CL External Signals Connector J4 to a bracket mounted DB25. Install the bracket assembly into an adjacent PC expansion slot and connect the free cable end to the **X64-CL J4** header. When connecting to J4, make sure that the cable pin 1 goes to J4 pin 1 (see layout drawings for your board revision: "X64-CL Layout Drawings" on page 82).

Warning: Proceed with caution when connecting external devices or other computers to the signals available on the external signals connector. Grounds should connect first and devices should be power up at the same time. External signal sources must not have voltage spikes or transients, else damage may occur on the X64-CL.

External Signals Connector Bracket Assembly Drawing


Pin 1
Flat cable 12"
DB25 Female Connector mounted on bracket
Header Connector 26-Pin Female Connect to J4
Note: Tracks marked as reserved are not connected to the Header Connector

External Signals Connector Bracket Assembly Signal Description

Refer to the table "J4 (CL or EM rev. A0, A1): External Signals Connector " on page 89 for important signal descriptions.

DB25 Pin Number	Signal Names	Connector (to J4)
1, 2, 3, 4, 17, 5, 18, 23, 12, 13	Reserved	1, 3, 5, 7, 8, 9, 10, 20, 23, 25
6	TrigIn 1+	11
19	TrigIn 1-	12
7	TrigIn 2+	13
20	TrigIn 2-	14
8	Shaft Encoder phase A +	15
21	Shaft Encoder phase A -	16
9	Shaft Encoder phase B +	17
22	Shaft Encoder phase B -	18
11	Strobe 1	21
10	Strobe 2	19
14, 15, 16, 24, 25,	Ground	2, 4, 6, 22, 24

J201 (X64-EM rev. A2 & later): External Signals Connector

Contact Teledyne DALSA for information on the External Signals Connector Bracket Assembly to bring out the signals from the X64-CL EM External Signals Connector J201 to a bracket mounted connector.

Warning: Proceed with caution when connecting external devices or other computers to the signals available on the external signals connector. Grounds should connect first and devices should be power up at the same time. External signal sources must not have voltage spikes or transients, else damage may occur on the X64-CL.

J201 Signal Descriptions

Pin #	Signal	X64-CL EM Description
1	Trig1+	External Trigger 1 + input *see note 1*
2	Trig1-	External Trigger 1 - input
3	Trig2+	External Trigger 2 + input
4	Trig2-	External Trigger 2 - input
5	PHA+	Shaft Encoder Phase A + or line trigger *see note 2 & 3 & 4*
6	PHA-	Shaft Encoder Phase A - *see note 7*
7	PHB+	Shaft Encoder Phase B + or line trigger
8	PHB-	Shaft Encoder Phase B - *see note 7*
9	STROBE2	Strobe control 2 (TTL)
10	Reserved	Reserved
11	STROBE1	Strobe control 1 (TTL) *see note 5*
12		Reserved
13		Reserved
14	GND	Ground
15	DC Power	Voltage selected (+12 or +5) via J8 *see note 6*
16		Reserved

Notes for X64-CL EM External Signals Connector

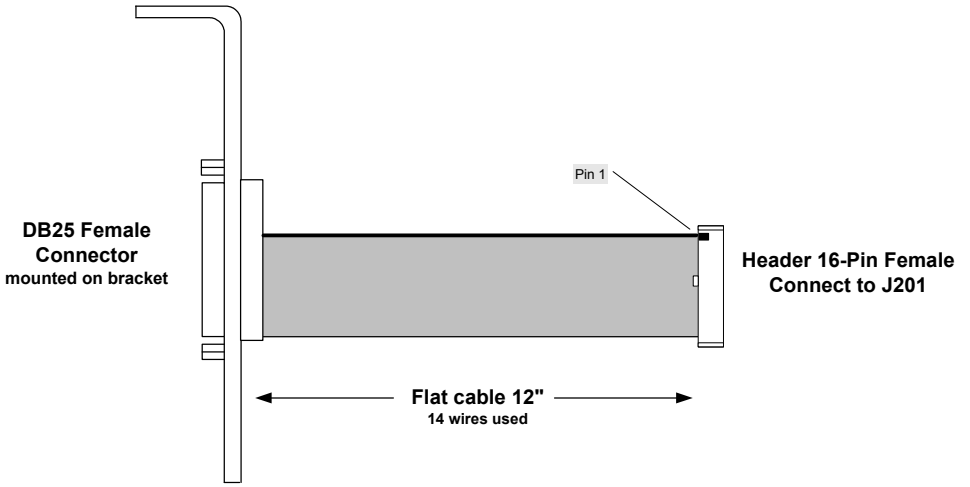
1. Refer to Sapera parameters
   CORACQ_PRM_EXT_TRIGGER_LEVEL
   CORACQ_PRM_EXT_FRAME_TRIGGER_LEVEL
   CORACQ_PRM_EXT_TRIGGER_ENABLE
   CORACQ_PRM_EXT_TRIGGER_DETECTION
2. See "Line Trigger Source Selection for Linescan Applications" on page 56 for more information.
   Refer to Sapera parameters CORACQ_PRM_SHAFT_ENCODER_ENABLE
   CORACQ_PRM_SHAFT_ENCODER_DROP
   or refer to CORACQ_PRM_EXT_LINE_TRIGGER_ENABLE
   CORACQ_PRM_EXT_LINE_TRIGGER_DETECTION
   CORACQ_PRM_EXT_LINE_TRIGGER_LEVEL (fixed at LVDS)
   CORACQ_PRM_EXT_LINE_TRIGGER_SOURCE
3. Important: When using only one shaft encoder input phase, say phase A, then the phase B inputs must be terminated by connecting phase B- to board ground available on any pin labeled GND and phase B+ to any DC source with a minimum of 100 mV positive relative to the phase B- input.
4. See for details on using a TTL shaft encoder signal.
5. Refer to Sapera parameters CORACQ_PRM_STROBE_ENABLE
   CORACQ_PRM_STROBE_POLARITY, CORACQ_PRM_STROBE_LEVEL,
   CORACQ_PRM_STROBE_METHOD, CORACQ_PRM_STROBE_DELAY
   CORACQ_PRM_STROBE_DURATION
6. The supplied host PC voltage is selected (+12 or +5) via the shorting jumper J8. A 1.5A resettable fuse is included on the board. If the fuse is tripped, power off the host computer to reset the fuse.
7. See "Connecting a TTL Shaft Encoder Signal to the LVDS Input" on page 101 for details on using a TTL shaft encoder signal.

X64-EM rev. A2 & later: External Signals Connector Bracket Assembly

Cable assembly **OC-64CC-GIO25EM, connected to J201**, brings trigger and strobe signals to a standard DB25 female connector. Install the bracket assembly into an adjacent PC expansion slot and connect the free cable end to the X64-CL J201 header. When connecting, make sure that the cable pin 1 goes to J201 pin 1 (see layout drawing "X64-CL Revision A2 Layout Drawing" on page 83).

Warning: Proceed with caution when connecting external devices or other computers to the signals available on the external signals connector. Grounds should connect first and devices should be power up at the same time. External signal sources must not have voltage spikes or transients, else damage may occur on the X64-CL.

External Signals Connector Bracket Assembly Drawing



External Signals Connector Bracket Assembly Signal Description

The following table defines the DB25 pinout.

Description	Pin Number	Pin Number	Description
Reserved	1	14	Reserved
Reserved	2	15	Reserved
Reserved	3	16	Reserved
Reserved	4	17	Reserved
Reserved	5	18	Reserved
External Trigger 1 + input	6	19	External Trigger 1 - input
External Trigger 2 + input	7	20	External Trigger 2 - input
Shaft Encoder Phase A + input	8	21	Shaft Encoder Phase A - input
Shaft Encoder Phase B + input	9	22	Shaft Encoder Phase B - input
Strobe control 2 (TTL output)	10	23	Reserved
Strobe control 1 (TTL output)	11	24	Reserved
Reserved	12	25	Ground
Reserved	13		

Hirose Connector Module Option (connects internally to J3)

The 12 pin Hirose connector option is available by special order. Contact sales at Teledyne DALSA for information about this factory installed option. **Important:** the pinout described below is not intended to connect to any camera on the market.

Warning: Proceed with caution when connecting external devices or other computers to the signals available on the Hirose connector. Grounds should connect first and devices should be power up at the same time. External signal sources must not have voltage spikes or transients, else damage may occur on the X64-CL.

Pin #	Signal	Description
1	Power	5V or 12V selectable with Jumper J8 for Camera Power (1.5A max)
2	GND	Ground for Camera
8	TrigIn 1+	TTL Trigger 1 In or LVDS Trigger 1 In + *TTL or LVDS is selected by the Sapera VIC parameter* External Trigger Level *or* External Frame Trigger Level *or the Sapera parameter* CORACQ_PRM_EXT_TRIGGER_LEVEL *or* CORACQ_PRM_EXT_FRAME_TRIGGER_LEVEL
7	TrigIn 1-	LVDS Trigger 1 In - (or TTL Trigger 1 In GND)
9	TrigIn 2+	TTL Trigger 2 In or LVDS Trigger 2 In +
6	TrigIn 2-	LVDS Trigger 2 In - (or TTL Trigger 2 In GND)
12	Phase A +	LVDS Shaft Encoder phase A + (or phase A TTL input +)
5	Phase A -	LVDS Shaft Encoder phase A – (see"Connecting a TTL Shaft Encoder Signal to the LVDS Input" on page 101 for details on using a TTL shaft encoder signal).
10	Phase B +	LVDS Shaft Encoder phase B + (or phase B TTL input +)
4	Phase B -	LVDS Shaft Encoder phase B –
11	Strobe 1	TTL Strobe 1 output : to connect to a strobe lamp
3	Strobe 2	TTL Strobe 2 output : to connect to a strobe lamp

Important: When using only one shaft encoder input phase with a X64-CL Full, say phase A, then the phase B inputs must be both terminated by connecting phase B- to board ground available on any pin labeled GND and phase B+ to any DC source with a minimum of +100 mV relative to ground.

LVDS Shaft Encoder Input Specifications

Dual Quadrature Shaft Encoder Inputs (phase A and phase B) are able to connect to differential signals (LVDS) or single ended TTL 5V source signals. The following figure is typical for each input.

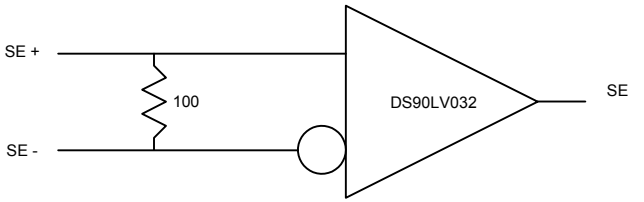


Figure 21: Shaft Encoder Input Electrical Diagram

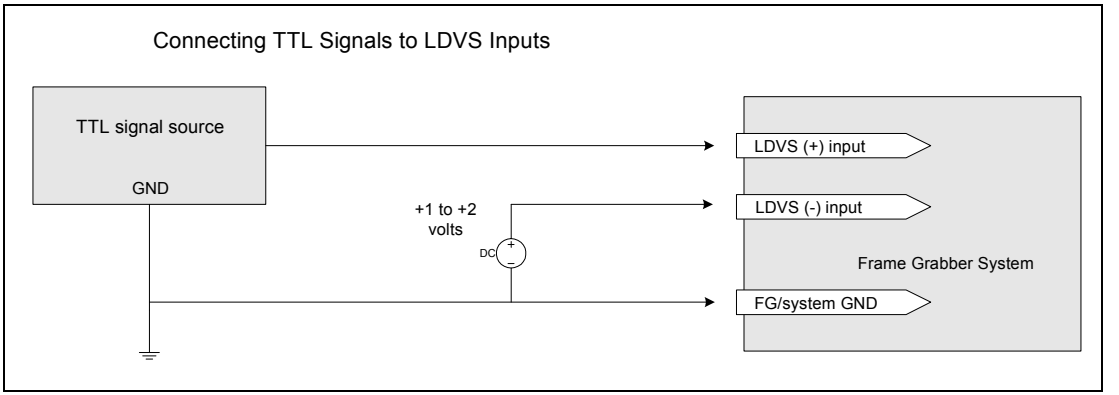
- The maximum frequency for any shaft encoder input is 1 MHz.
- See "Line Trigger Source Selection for Linescan Applications" on page 56 for more information.
- Refer to Sapera parameters:
  CORACQ_PRM_SHAFT_ENCODER_ENABLE CORACQ_PRM_SHAFT_ENCODER_DROP
  or refer to CORACQ_PRM_EXT_LINE_TRIGGER_ENABLE
  CORACQ_PRM_EXT_LINE_TRIGGER_DETECTION
  CORACQ_PRM_EXT_LINE_TRIGGER_LEVEL (fixed at RS-422)
  CORACQ_PRM_EXT_LINE_TRIGGER_SOURCE

- See also *.cvi file entries:
  Shaft Encoder Enable, Shaft Encoder Pulse Drop, Shaft Encoder Source
  or see External Line Trigger Enable, External Line Trigger Detection, External Line Trigger Level, External Line Trigger Source.

Connecting a TTL Shaft Encoder Signal to the LVDS Input

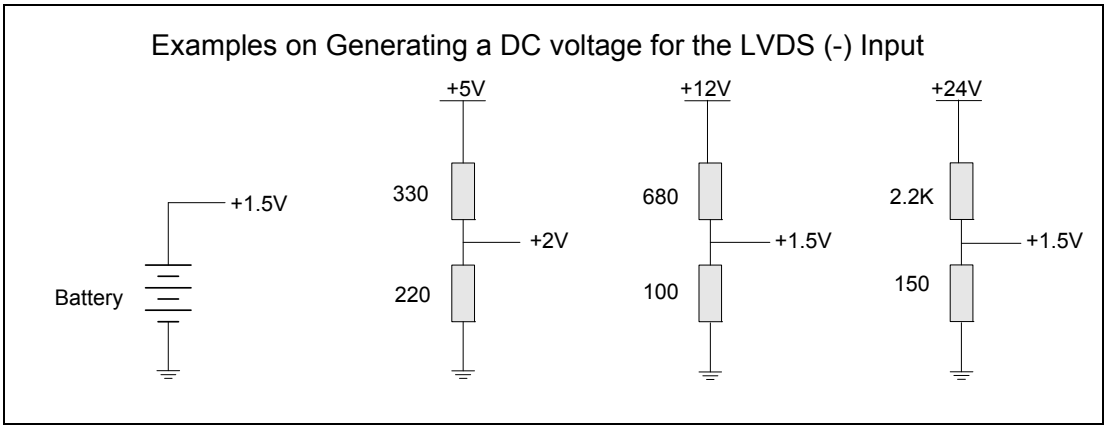
A TTL shaft encoder signal can be directly connected to the X64-CL LVDS (+) input but the low side (-) input of the pair must be biased with a DC voltage to ensure reliable operation. This section shows the connection diagram along with suggestions as to how to generate the DC bias voltage. The actual physical wiring is left as an additional detail to interfacing a shaft encoder to the X64-CL to the imaging system.

TTL Shaft Encoder to LVDS Input Block Diagram



- LVDS (-) input is biased to a DC voltage from +1 to +2 volts.
- This guarantees that the TTL signal connected to the LVDS (+) input will be detected as a logic high or low relative to the (-) input.
- The TTL shaft encoder ground, the bias voltage ground, and the X64-CL computer system ground must be connected together.

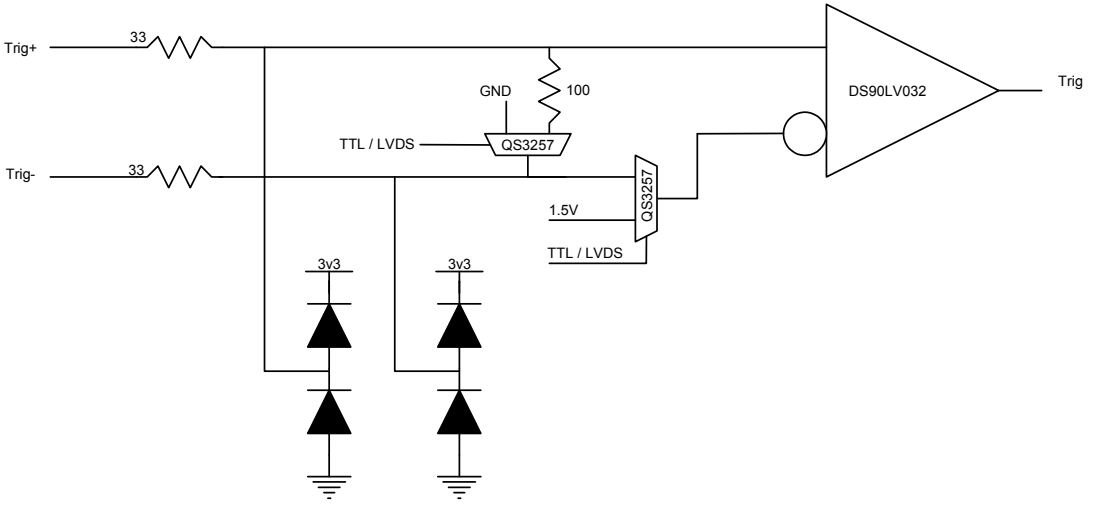
LVDS (-) Input Bias Source Generation



- DC voltage for the LVDS (-) input can be generated by a resister voltage divider.
- Use a single battery cell if this is more suitable to your system.
- A DC voltage (either +5 or +12) is available on External Signals Connector J4. See "J8 (CL), J12 (CL EM): Power to Camera Voltage Selector" on page 105 for information.

External Trigger TTL Input Electrical Specification

- The incoming trigger pulse is “debounced” to ensure that no voltage glitch would be detected as a valid trigger pulse. This debounce circuit time constant can be programmed from 0µs to 255µs. Any pulse smaller than the programmed value is blocked and therefore not seen by the acquisition circuitry.
- The input trigger can be software selected as LVDS or TTL. In LVDS Mode, the switches are set to connect the “-“pin to the “-“input of the LVDS receiver with a 100ohm termination to the “+”pin. In TTL Mode, the switches are set to disconnect the 100ohm termination and to put 1.5V to the “-“input of the LVDS receiver.



Electrical parameters	Description	Value
TrigIn low	Low logic level input	<= 0.8 V
TrigIn high	High logic level input	>= 2.0 V (max = 5 V)
TrigIn pulse width	Minimum trigger pulse width	100 ns

Sapera parameters for External Trigger:

CORACQ_PRM_EXT_TRIGGER_ENABLE = CORACQ_VAL_EXT_TRIGGER_ON

CORACQ_PRM_EXT_TRIGGER_SOURCE

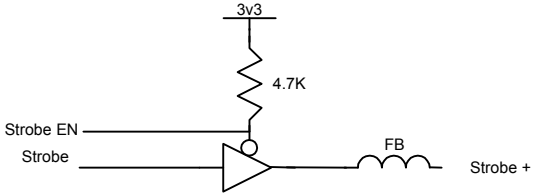
CORACQ_PRM_EXT_TRIGGER_DETECTION = {CORACQ_VAL_RISING_EDGE, CORACQ_VAL_FALLING_EDGE, CORACQ_VAL_ACTIVE_LOW. CORACQ_VAL_ACTIVE_HIGH}

CORACQ_PRM_EXT_TRIGGER_DURATION: Debounce duration

CORACQ_PRM_EXT_TRIGGER_LEVEL

Strobe TTL Output Electrical Specification

The strobe output is done with a TTL driver at 3.3V. The output is enabled/disabled by software control. It is generated by a SN74LVC1G125 device.



Electrical parameters	Description	Value
$V_{OH\ typ}$	Typical high-level output voltage	3.9V
$I_{OH\ max}$	Maximum high-level output current	-8mA (sourcing)
$I_{OL\ max}$	Maximum low-level output current	8mA (sinking)

Sapera parameters for Strobe :

Refer to Strobe Method in Sapera documentation

CORACQ_PRM_STROBE_ENABLE = TRUE

CORACQ_PRM_STROBE_METHOD={CORACQ_VAL_STROBE_METHOD_1, CORACQ_VAL_STROBE_METHOD_2}

CORACQ_PRM_STROBE_POLARITY={CORACQ_VAL_ACTIVE_LOW, CORACQ_VAL_ACTIVE_HIGH}

CORACQ_PRM_STROBE_DELAY: Pulse offset from trigger event

CORACQ_PRM_STROBE_DELAY_2: Duration of exclusion region

CORACQ_PRM_STROBE_DURATION: Pulse duration

J8 (CL), J12 (CL EM): Power to Camera Voltage Selector

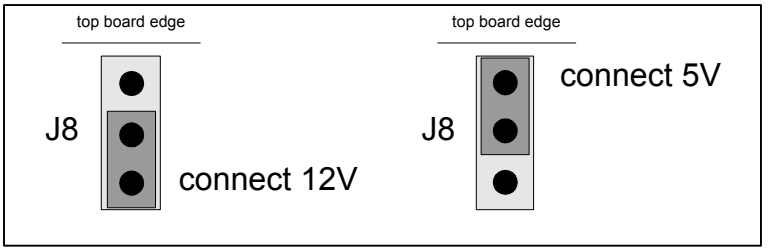
When the PC floppy drive power supply cable is connected to J9/J13, a shorting jumper on J8/J12 selects either 5 Vdc or 12 Vdc for the camera power supply. This supply voltage is available on the Hirose connector module or on the External Signals Connector block.

Important:

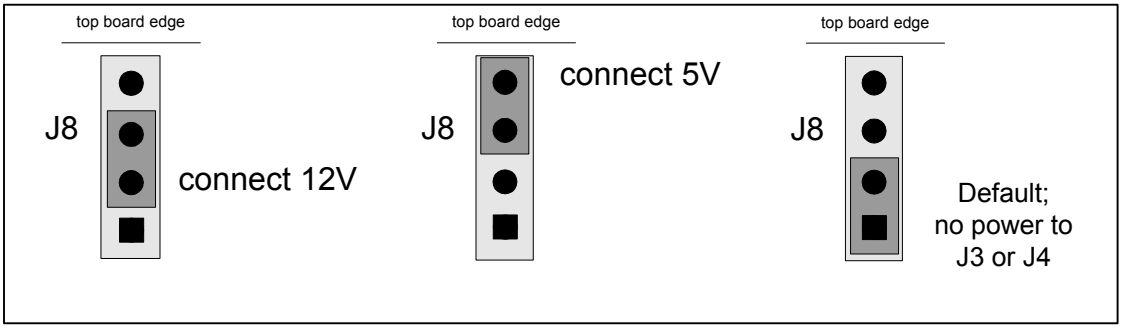
For **X64-CL revision A2, A3, A4** boards, the selected voltage is *not* fused. It is strongly recommended that any cabling using the DC power has an inline fuse (1.5A max).

For **X64-CL revision A5 and later** boards, or **X64-CL EM A2** boards, a 1.5A resettable fuse is included on the board. If the fuse is tripped, power off the host computer to reset the fuse.

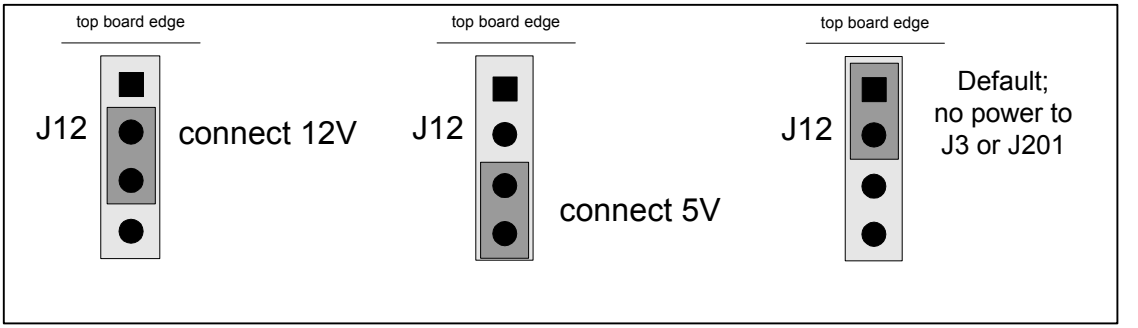
J8 as on X64-CL revision A1 and A2 boards



J8 as on X64-CL revision A3, A4, A5, A6, A7, A8 boards



J12 as on X64-CL EM revision A2 board



J9 (CL), J13 (CL EM): PC Power to Camera Interface

Connect the PC floppy drive power connector to J9/J13 so as to supply DC power to the camera. Place the J8/J12 shorting jumper so as to select 5 Vdc or 12 Vdc for the camera.

J11 (CL), J14 (CL EM): Start Mode

- Default Mode: Shunt jumper is installed.
- Safe Mode: Shunt jumper is removed if any problems occurred while updating the X64 firmware. With the jumper off, reboot the computer and update the firmware again. When the update is complete, install the jumper and reboot the computer once again. (See "Recovering from a Firmware Update Error" on page 26).

J7, J12 (CL): Reserved

J7, J9, J10, J11, J15, J200, J202 (CL EM): Reserved

Brief Description of Standards RS-232, RS-422, & RS-644 (LVDS)

RS-232

Short for ***r**ecommended **s**tandard-232C,* a standard interface approved by the Electronic Industries Association (EIA) connecting serial devices.

The standards for RS-232 and similar interfaces usually restrict RS-232 to 256kbps or less and line lengths of 15M (50 ft) or less.

Transmitted Data (TxD) This signal is active when data is transmitted from the DTE device to the DCE device. When no data is transmitted, the signal is held in the mark condition (logic '1', negative voltage).

Received Data (RxD) This signal is active when the DTE device receives data from the DCE device. When no data is transmitted, the signal is held in the mark condition (logic '1', negative voltage).

DTE (Data Terminal Equipment)
DCE (Data Communication Equipment)

RS-422

RS-422 uses a twisted-pair wire (i.e., 2 wires) for each signal. The differential drive voltage swing is 0 to +5V. RS-422 does not have tri-state capability (its driver is always enabled) and it is therefore usable only in point-to-point communications.

Although RS-422 is noise resistant, due to being differential data can still be damaged by EMI/RFI. A shielded cable can protect the transmitters/receivers from EMI/RFI.

RS-644 (LVDS)

LVDS (Low-Voltage Differential Signaling): method to communicate data using a very low voltage swing (about 350mV) over two differential PCB traces or a balanced cable. LVDS allows single channel data transmission at hundreds of Megabits per second (Mbps).

Camera Link Interface

Camera Link Overview

Camera Link is a communication interface for vision applications developed as an extension of National Semiconductor's Channel Link technology. The advantages of the Camera Link interface are that it provides a standard digital camera connection specification, a standard data communication protocol, and simpler cabling between camera and frame grabber.

The Camera Link interface simplifies the usage of increasingly diverse cameras and high signal speeds without complex custom cabling. For additional information concerning Camera Link, see

http://en.wikipedia.org/wiki/Camera_Link.

Rights and Trademarks

Note: The following text is extracted from the Camera Link Specification 1.1 (January 2004).

The Automated Imaging Association (AIA), as sponsor of the Camera Link committee, owns the U.S. trademark registration for the Camera Link logo as a certification mark for the mutual benefit of the industry. The AIA will issue a license to any company, member or non-member, to use the Camera Link logo with any products that the company will self-certify to be compliant with the Camera Link standard. Licensed users of the Camera Link logo will not be required to credit the AIA with ownership of the registered mark.

3M™ is a trademark of the 3M Company.

Channel Link™ is a trademark of National Semiconductor.

Flatlink™ is a trademark of Texas Instruments.

Panel Link™ is a trademark of Silicon Image.

Data Port Summary

The Camera Link interface has three configurations. A single Camera Link connection is limited to 28 bits requiring some cameras to have multiple connections or channels. The naming conventions for the three configurations are:

- Base: Single Channel Link interface, single cable connector.
- Medium: Two Channel Link interface, two cable connectors.
- Full: Three Channel Link interface, two cable connectors.

A single Camera Link port is defined as having an 8-bit data word. The "Full" specification supports 8 ports labeled as A to H.

Camera Signal Summary

Video Data

Four enable signals are defined as:

- FVAL Frame Valid (FVAL) is defined HIGH for valid lines.
- LVAL Line Valid (LVAL) is defined HIGH for valid pixels.
- DVAL Data Valid (DVAL) is defined HIGH when data is valid.
- Spare A spare has been defined for future use.

All four enables must be provided by the camera on each Channel Link. All unused data bits must be tied to a known value by the camera.

Camera Controls

Four LVDS pairs are reserved for general-purpose camera control, defined as camera inputs and frame grabber outputs.

- Camera Control 1 (CC1)
- Camera Control 2 (CC2)
- Camera Control 3 (CC3)
- Camera Control 4 (CC4)

Note: the X64-CL by default implements the control lines as follows
(using Teledyne DALSA Corporation terminology).

(CC1) EXYNC
(CC2) PRIN
(CC3) FORWARD
(CC4) HIGH

Communication

Two LVDS pairs have been allocated for asynchronous serial communication to and from the camera and frame grabber. Cameras and frame grabbers should support at least 9600 baud.

- SerTFG Differential pair with serial communications to the frame grabber.
- SerTC Differential pair with serial communications to the camera.

The serial interface protocol is one start bit, one stop bit, no parity, and no handshaking.

Camera Link Cables

For additional information on Camera Link cables and their specifications, visit the following web sites:

3 M	http://www.3m.com/interconnects / *(enter Camera Link as the search keyword)*
Nortech Systems	http://www.nortechsys.com/intercon/CameraLinkMain.htm

Contact Information

Sales Information

Visit our web site: www.teledynedalsa.com/mv

Email: mailto:info@teledynedalsa.com

Canada	USA Sales
Teledyne DALSA — Montreal office 7075 Place Robert-Joncas, Suite #142 St. Laurent, Quebec, Canada H4M 2Z2 Tel: (514) 333-1301 Fax: (514) 333-1388	Teledyne DALSA — Billerica office 700 Technology Park Drive Billerica, Ma. 01821 Tel: (978) 670-2000 Fax: (978) 670-2010

Asia Sales	European Sales
Teledyne DALSA Asia Pacific Ikebukuro East 13F 3-4-3 Higashi Ikebukuro, Toshima-ku, Tokyo Japan Tel: +81 3 5960 6353 Fax: +81 3 5960 6354	Teledyne DALSA Europe Breslauer Str. 34 D-82194 Gröbenzell (Munich) Germany Tel: +49 - 8142 – 46770 Fax: +49 - 8142 - 467746

Technical Support

Submit any support question or request via our web site:

Technical support form via our web page: Support requests for imaging product installations, Support requests for imaging applications	http://www.teledynedalsa.com/mv/support
Camera support information	
Product literature and driver updates	

Glossary of Terms

Bandwidth

Describes the measure of data transfer capacity. PCI devices must share the maximum PCI bus bandwidth when transferring data to and from system memory or other devices.

CAM

Sapera camera file that uses the file extension CCA by default. Files using the CCA extension, also called CAM files (CORECO CAMERA files), contain all parameters which describe the camera video signal characteristics and operation modes (i.e. what the camera outputs).

Channel

Camera data path that includes all parts of a video line.

Checksum

A value used to ensure data is stored without error. It is created by calculating the binary values in a block of data using some algorithm and storing the results with the data.

CMI

Client **M**odification **I**nstruction. A client requested engineering change applied to a Teledyne DALSA board product to support either a non-standard function or custom camera.

Contiguous memory

A block of physical memory, occupying consecutive addresses.

CRC

Proprietary Sapera raw image data file format that supports any Sapera buffer type and utilizes an informative file header. Refer to the *Sapera Basic Modules Reference Manual* "Buffer File Formats" section.

Firmware

Software such as a board driver that is stored in nonvolatile memory mounted on that board.

Frame buffer

An area of memory used to hold a frame of image data. A frame buffer may exist on the acquisition hardware or be allocated by the acquisition hardware device driver in host system memory.

Grab
Acquiring an image frame by means of a frame grabber.

Host
Refers to the computer system that supports the installed frame grabber.

Host buffer
Refers to a frame buffer allocated in the physical memory of the host computer system.

LSB
Least Significant Bit in a binary data word.

MSB
Most Significant Bit in a binary data word.

PCI 32
Peripheral Component Interconnect. The PCI local bus is a 32-bit high-performance expansion bus intended for interconnecting add-in boards, controllers, and processor/memory systems.

PCI 64
A superset of the PCI specification providing a 64 bit data path and a 66 MHz clock.

Pixel
Picture Element. The number of pixels describes the number of digital samples taken of the analog video signal. The number of pixels per video line by the number of active video lines describes the acquisition image resolution. The binary size of each pixel (i.e., 8-bits, 15-bits, 24-bits) defines the number of gray levels or colors possible for each pixel.

RAW
A Sapera data file format where there is no header information and that supports any Sapera buffer type. Refer to the *Sapera Basic Modules Reference Manual* "Buffer File Formats" section.

RISC
(**R**educed **I**nstruction **S**et **C**omputer) A computer architecture that reduces chip complexity by using simpler instructions.

Scatter Gather

Host system memory allocated for frame buffers that is virtually contiguous but physically scattered throughout all available memory.

Tap

Data path from a camera that includes a part of or whole video line. When a camera tap outputs a partial video line, the multiple camera tap data must be constructed by combining the data in the correct order.

VIC

Sapera camera parameter definition file that uses the file extension CVI by default. Files using the CVI extension, also know as VIC files (CORECO VIDEO files), contain all operating parameters related to the frame grabber board (i.e. what the frame grabber can actually do with camera controls or incoming video).

Index

1

10 tap formats 65

A

Acquisition and Control Unit 61
Acquisition events 62
acquisition module 62
acquisition parameters 49
administrator 11, 19
AUTORUN 12

B

Block Diagram 51
BoardInfo.txt 27
boot recovery mode 26

C

camera configuration file 35
camera control 7, 20, 89
Camera file 49, 58, 60
Camera Link 7, 9, 85, 89, 109
Camera Link cabling 20
Camera Link control 89
camera timing 35
CamExpert 49, 58, 60
CamExpert parameters 36
Certifications 81
communication ports 9
configuration switch 26
Contiguous Memory 24
CORACQ_PRM_EXT_LINE_TRIGGER_DETECTION 91, 96, 100
CORACQ_PRM_EXT_LINE_TRIGGER_ENABLE 91, 96, 100
CORACQ_PRM_EXT_LINE_TRIGGER_LEVEL 91, 96, 100
CORACQ_PRM_EXT_LINE_TRIGGER_SOURCE 91, 96, 100
CORACQ_PRM_EXT_TRIGGER_DETECTION 91, 96
CORACQ_PRM_EXT_TRIGGER_ENABLE 91, 96
CORACQ_PRM_EXT_TRIGGER_LEVEL 91, 96
CORACQ_PRM_SHAFT_ENCODER_DROP 91, 96, 100
CORACQ_PRM_SHAFT_ENCODER_ENABLE 91, 96, 100
CORACQ_PRM_SHAFT_ENCODER_LEVEL 91, 96, 100
CORACQ_PRM_STROBE_DELAY 92, 96
CORACQ_PRM_STROBE_DURATION 92, 96
CORACQ_PRM_STROBE_ENABLE 92, 96
CORACQ_PRM_STROBE_LEVEL 92, 96
CORACQ_PRM_STROBE_METHOD 92, 96
CORACQ_PRM_STROBE_POLARITY 92, 96

D

DALSA Device Manager 27
DALSA Imaging drivers 30
Data Overflow event 62
debounce circuit time constant 102
debounce control 61
default firmware 64
Device Manager 13
double buffering memory 33
driver upgrade 19

E

End of Frame event 62
End of Transfer event 63
error notification 61
External Signals Connector 55, 57, 58, 59, 86, 89, 93, 94, 95
External Signals Connector Bracket Assembly 58, 93, 97
external trigger source 56

F

failure - firmware upgrade 26
Firmware Loader 13
firmware revision 23
firmware selection 7
Found New Hardware Wizard 11
frame buffer 7, 25, 59
Frame Lost event 62
Frame Sync 60
FRAME_RESET 59

H

half length PCI 80
HTML help 6
HyperTerminal 9, 20

I

image processing 6
installer response file 16

J

jumpers 86, 105

L

launch.exe 12
Line Scan 8, 58
Log Viewer program 31
LUT availability 64
LVDS pairs 89
LVDS Trigger In 99

M

MDR-26 85

N

National Semiconductor 109
non-standard cameras 80

O

OC-64CC-0TIO1 58, 93
out-of-memory error 25

P

PCI bus latency 61
PCI Bus Number 29
PCI configuration space 27, 29, 34
PCI conflict 27
PDF 6
Phase A 58
Phase B 58
Physical Dimensions 80
Power Requirements 80

Q

Quadrature-Shaft-Encoder 9

S

Sapera buffers allocation 24
Sapera CamExpert 34
Sapera CD-ROM 12, 19
Sapera configuration program 21, 22, 24
Sapera LT Development Library 11
Sapera LT User's manual 12
Sapera messaging 24
scatter gather buffers 25
Scatter-Gather 9
serial communication port 20
shaft encoder 7, 9, 58
shaft encoder source 56
software trigger 34
Specialty Cameras 80
Static electricity 11
system COM port 20

T

technical support 19, 23, 30, 33
transfer module 62
trigger 9, 58, 59
trigger signal 61
TTL shaft encoder 101
TTL Trigger In 99

V

viewer program 23, 31
virtual frame buffer 59
visual LED indicators 9

W

Web inspection 58
Windows HyperTerminal 20
Windows operating system memory 25
workstation 11, 19, 20

X

X64-CL connector bracket 85
X64-CL serial port 20, 22
X64-CL Viewer report 23